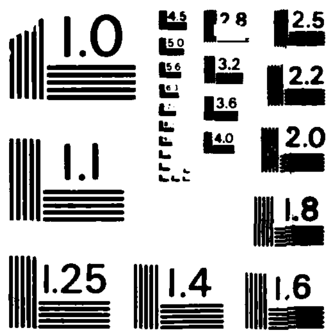
AD-A179 422 BASING THE US AIR FORCE SPECIAL OPERATIONS FORCES(U) 1/2
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING M E KRAUS DEC 86 AFIT/GOR/OS/86D-6

UNCLASSIFIED F/G 15/5 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AFIT/GOR/OS/86D-6

BASING THE US AIR FORCE

SPECIAL OPERATIONS FORCES

THESIS

Mark E. Kraus
Captain, USAF

AFIT/GOR/OS/86D-6

AFIT/GOR/OS/86D-6

BASING THE US AIR FORCE SPECIAL OPERATIONS FORCES

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

QUALITY INSPECTED 2

Mark E. Kraus

Captain, USAF

December 1986

Approved for public release; distribution unlimited.

Preface

This study modified the existing Hq MAC Combat Rescue and Special Operations Forces Model (CRASOF) to include a multiple basing capability that it did not have. It then demonstrated the use of that model in assessing the relative capabilities of a force under different basing strategies. As a demonstration, this study does not involve any real basing problem. The scenario used was dreamed up by me, and is far enough from reality to be safely unclassified. Studies run with actual scenarios and data would be classified. The model is not included in this thesis because of its size, but is available from Hq MAC/XPCS, for whom it was modified.

At this point, I would like to thank several people for their help in this effort. First, thanks to Maj Jack R. Dickinson, who built the original CRASOF model with my help and recommended this thesis. His inspiration and patience in teaching me is very appreciated. Second, his successor at Hq MAC, Capt Joseph Neimeyer, helped by providing data and by acting as a sounding board throughout the work. I would also like to thank LtC Skip Valusek, my advisor, for his support and criticism.

Most importantly, I would like to thank my wife, Dee, who helped me through the ups and downs of this work. Without her active support, this thesis would be a product to be put on the back shelf somewhere and forgotten as no practical use to anyone.

Mark E. Kraus

Table of Contents

	Page
Preface	ii
List of Figures	v
List of Tables	vi
Abstract	vii
I. Problem Formulation	1
Introduction	1
Research Question	2
Research Objectives	2
Scope	2
Alternative Generation Review	3
Location Analysis Literature Review	5
Methodology Overview	9
Summary	10
II. System Modeling	11
Introduction	11
US Army Special Forces Airlift Requirements	11
US Air Force Activities	17
Summary	24
III. Model Coding and Testing	25
Introduction	25
History	25
Modifications and Additions	26
Model Verification	28
Model Validation	30
Summary	31
IV. Scenario and Experimental Design	32
Scenario	32
Factors of Interest	33
Experimental Design	34
Sample Size	35
Variance Reduction Techniques	36
Summary	36

V. Analysis of Results 38

Introduction 38
Technical Description 38
Regional Priorities 59
Interpretation of Results 59
Mission Delays 61
Quick Response Use of the Model 63
Summary 64

VI. Observations and Recommendations 66

Introduction 66
Observations 66
Recommendations for Further Study 69

Appendix A: Programmer's Manual 71

Introduction 72
Network Description 77
Attribute Definitions 78
Program MAIN 80
ALLOC 81
ARCCOS 88
ARSPOT 89
ARTANK 91
COMBLK 94
DELAY 95
DISTANCE 96
EVENT 97
INTLC 104
LASTAR 106
LOSSES 107
MAKECR 109
NEWAR 110
OTPUT 114
PLANE 115
SPARE 116
STATS 117
TARGET 118
USERF 119
WABORT 122
WXCELL 123

Appendix B: SLAM Code 124

Appendix C: Input Data Files 131

Bibliography 149

Vita 150

List of Figures

Figure		Page
1.	Normal Probability Plot for Total Successful Missions	45
2.	Normal Probability Plot for Successful Infiltrations	46
3.	Normal Probability Plot for Successful Exfiltrations	47
4.	Normal Probability Plot for Successful Resupplies	48
5.	Normal Probability Plot for Infiltration Delay	49
6.	Normal Probability Plot for Exfiltration Delay	50
7.	Normal Probability Plot for Resupply Delay	51
8.	Residuals for Total Successful Missions	52
9.	Residuals for Successful Infiltrations	53
10.	Residuals for Successful Exfiltrations	54
11.	Residuals for Successful Resupplies	55
12.	Residuals for Infiltration Delay	56
13.	Residuals for Exfiltration Delay	57
14.	Residuals for Resupply Delay	58
15.	Calculating Inversions	60

List of Tables

Table		Page
I.	Sample Target Distribution	13
II.	Mission Rates	14
III.	A-team Availability	16
IV.	Aircraft Capabilities	19
V.	Aircraft and Aircrew Availability	22
VI.	Asset Location	23
VII.	Target Distribution	32
VIII.	Mission and A-team Availability Rates	33
IX.	USAF Assets Available	33
X.	Basing Options	35
XI.	ANOVA for Successful Missions	39
XII.	ANOVA for Successful Infiltrations	39
XIII.	ANOVA for Successful Exfiltrations	40
XIV.	ANOVA for Successful Resupplies	40
XV.	ANOVA for Infiltration Delay	41
XVI.	ANOVA for Exfiltration Delay	41
XVII.	ANOVA for Resupply Delay	42
XVIII.	Missions Accomplished	43
XIX.	Mission Delay (Days)	43
XX.	Ranking of Basing Options	44
XXI.	Realized Regional Priorities	60
XXII.	Ranking of Excursions	62
XXIII.	Quick Response Basing Option	63
XXIV.	Files Used	131

Abstract

The Air Force Special Operations Forces (AFSOF) are currently overtasked. The only existing Hq MAC AFSOF model is limited by the assumption that all AFSOF assets are colocated. This research effort was directed toward removing this limitation so the model could be used to address basing questions. The model was modified to include geographical locations rather than just distances. The target data base was modified, and a basing data base was added.

The model was then demonstrated using a representative scenario and representative data. The scenario and data were coordinated with Hq MAC to insure they were of the right form but not close to any classified information. The study involved three basing options to be compared for mission accomplishment. The options were compared for total successful missions, which was also broken down by mission type; average mission delay by mission type; and how well they implemented the desired regional priority scheme.

The uses and limitations of the model as well as potential areas of improvement were discussed. A major limitation of the model is its restriction to use for long range planning. A look was taken at a deterministic model that could provide a short time response. It appeared feasible to use location/allocation methods. Such a model could be used in a decision support system to provide real time help in basing the AFSOF.

BASING THE US AIR FORCE
SPECIAL OPERATIONS FORCES

I. Problem Formulation

Introduction

The Air Force Special Operations Forces (AFSOF) are tasked with providing airlift support for Army Special Forces teams (A-teams). Three types of support provided are infiltration, or putting a team in place; exfiltration, or extracting a team; and resupply. The Army has planned A-team missions based on their target lists. However, the planned requirements for support are much larger than the AFSOF's capacity. This shortfall in capability is so large that it will be only partially met through funded aircraft procurement and modification programs. Funds are not available to meet the rest of the shortfall. Therefore, only part of the Army requirement will be met. The Military Airlift Command (MAC) has the responsibility for the AFSOF, and must decide how to base its forces to meet as much of the Army's requirement as possible. Currently this is done heuristically by the long range planners at Hq MAC. The first step is to form a set of politically and fiscally feasible basing alternatives. Then the best of these alternatives is chosen. However, there is no quantitative model used to help choose a best basing strategy from a set of feasible alternatives. No model exists to judge the relative capability of the alternatives. The Combat Rescue and Special Operations Forces simulation model (CRASOF) that Hq MAC has now is not adequate to evaluate basing strategies be-

cause it implicitly assumes that all aircraft are located at the same base. Since no current effectiveness model exists, choices between basing strategies must be based on the unaided personal judgments of the planners.

Research Question

Given a SOF airlift requirement and a limited number of airlift assets, which of several basing options will allow the AFSOF to meet the greatest portion of the US Army requirement?

Research Objectives

This research effort will develop a method to help Hq MAC long range planners to choose the basing strategy from among several possibilities that maximizes the portion of the Army requirement that can be met. In order to meet this objective, the following questions must first be answered.

Alternative Generation. How are feasible basing options generated at Hq MAC?

Model Development. What type of model is best suited to answer this question?

Data Availablility. What data are necessary and in what form are they available?

Scope

This research will consider only force capability as a criterion for the choice of basing strategy. Other criteria, such as cost, are assumed to be considered in generating the set of possible choices.

This is because the purpose of the model developed here is to provide input to the decision maker regard'ng force capability, which will be only one of the factors considered in making a final decision regarding basing. The primary measure of force capability will be the number of missions accomplished. A secondary measure will be the timeliness of those missions, or the average delay between mission tasking and flying the mission. A third measure will be the adherence to the desired regional priorities. The model will be designed to aid long range planners. As such, it will not be useful in short term planning or in other time critical situations.

Alternative Generation Review

The following description of the process used by Hq MAC to generate feasible basing alternatives is taken from an interview with LtC Paul J. Capicik, Hq MAC/XONP (1).

Several factors must be considered when choosing bases for use as wartime AFSOF beddown locations. They include the target locations, expected threat, expected threat avoidance methods, airfield requirements, logistics and intelligence support availability, and political considerations. These factors must all be considered and trade-offs made before a base is accepted as a candidate beddown location.

The first step is to lay out the expected targets and to identify all bases within a maximum allowable radius of the target areas. The maximum radius is a function of the types of aircraft to be used, their speeds, refueling capabilities, and crew time limitations. This provides a first estimate of the available candidate bases.

The next step is to consider the expected threat and the possible ingress and egress routes. For instance, if the threat is such that aircraft must fly around it to the north and approach from that direction only, basing aircraft to the south of the threat would require that they fly longer routes to get north of the threat. Therefore, a base to the south that is actually closer to the targets may well be farther away in flying distance. This step may eliminate bases that would require very long legs before beginning an ingress.

Next, to further reduce the number of candidate bases, eliminate any bases that do not meet the minimum airfield and support requirements. For example, a base with a runway that would accommodate only small planes but not a C-130 would be inappropriate for basing Talons and should not be considered as a possible beddown location for the MC-130. However, it could still be considered as a possible helicopter beddown location. Bases without potential logistics and intelligence support capability must also be dropped from consideration. A remote base may be ideal in every sense except it cannot be readily resupplied or it cannot get current intelligence data. That base must be eliminated because the success of a SOF mission is dependent on available logistics support and current intelligence.

Another important requirement in SOF basing is the need for deception in SOF. Any base to be used for SOF must be located so the missions can be undertaken with some deception concerning their targets and/or missions. If the base is located so there is only one target region that aircraft can reach from it, that base may need to be eliminated if missions to that region require deception.

Possibly the most important consideration in choosing bases is the political climate. AFSOF missions will start before the formal declaration of war, so the host countries must approve the use of their bases before the missions can be flown. The ideal base may be located in a country which will allow no hostile missions to be flown from its territory before a formal declaration of war. Such a base cannot be used by the AFSOF initially, because AFSOF operations would be expected to start before a declaration of war. It may be available after a formal declaration of war, but it cannot be considered for pre-war actions. A good example of this is the 1986 airstrike against Libya which was launched from England. Spanish basing may have been preferable, but the political climate would not allow it. Therefore, bases in Spain could not be considered as possible launching or recovery points for the mission.

Location Analysis Literature Review

The problem of choosing a basing strategy for locating aircraft is a specific application of the general location problem. The goal in location problems is to locate service facilities to minimize some cost function or to maximize the amount of the demand for that service that can be met. Location problems are generally modeled as networks. Service facilities and demand points are located at nodes or along arcs, and the costs are modeled as the arc lengths. Variables in the model are the demand for service, costs, and service facility availability. In deterministic problems, all three are known with certainty, while in stochastic problems, at least one of the three has a probability distribution associated with it.

Solution Techniques. Solution techniques have been developed to

solve both deterministic and stochastic problems. The techniques developed were based on the assumption that all demand for the service must originate at nodes in the network (2:49). Many of the techniques, especially those designed to solve stochastic models, also assume that the facilities are also located at nodes. This assumption is made to ensure the computational feasibility of the problem (2:49). In most cases, optimizing techniques were developed that guarantee a solution but were too bulky to solve in a cost effective manner with today's technology. Therefore, heuristic methods that provide good answers at a fraction of the cost of the optimizing techniques have been developed (8:178)

Optimizing Techniques. Linear programming is the most straightforward of the methods used for optimizing location problems. The objective is to optimize a linear cost function subject to the constraints describing available service. The major drawback of these techniques is their size. For example, a stochastic network with 50 nodes and 10 states (a moderately sized problem) would require over 25,500 equations to fully describe the system. Such a formulation would tax the largest existing computers (8:172). The usual method of dealing with such a large problem is to set up the problem and then use heuristic methods to solve it.

Another optimizing technique used is Lagrangian optimization. This method results in a much smaller mathematical formulation than linear programming, but is more difficult to solve. The system is again described by a system of equations - possibly not linear. The constraints are then combined with the cost function to form an unconstrained op-

timization problem. The problem is converted into its dual, and both the original Lagrangian and its dual are optimized (9:67). Although the Lagrangian method yields a smaller problem formulation, it can also be extremely unwieldy. Therefore, it is sometimes used to approximate a solution from which to start using heuristic methods (6:894).

Heuristic Techniques. Since exact methods for finding optimum solutions to location problems are very bulky, heuristic methods have been developed to provide quick solutions that are close enough to the optimum to be acceptable. The problem with testing heuristic methods is that their relative accuracy cannot be measured unless the optimum is known, in which case the heuristic method is not needed. The heuristic methods discussed here have been demonstrated in cases where the optimum solutions are known, so they have been accepted as good methods.

One-at-a-time exchange techniques are simple swapping methods. After a problem is formulated and the cost function is generated, an initial guess is made at the optimum solution. This initial guess may be to put all the facilities at the same node or at different nodes, or it may be generated using a subproblem formulated as a linear program or a Lagrangian problem. Then one location in the current solution is exchanged with a location not in the current solution if it decreases the overall cost. Only one swap occurs at a time. This continues until no single exchange will yield a better solution (8:172). The major drawback of this technique is that "no matter how well it does on the average, one has no way of knowing how well it does in a particular case" (8:172). In practice, however, the time required to get a 'solution' with this method is several orders of magnitude faster than the

optimizing techniques mentioned above (8:178).

Covering set methods use a different approach to solving location problems. The other methods use a mathematical programming approach, while covering set methods use a set theory approach. In set theory, a point is covered by a set if it is contained in that set. In this application, a point is a demand point, and the sets are formed by surrounding the candidate facility locations with circles of a given size. The size of each circle is determined by the maximum distance between a facility and a demand point it may cover. Alternate approaches used in covering set techniques are 1) to fix the maximum distance and find the number and locations of facilities needed, and 2) to fix the number of facilities and find the locations yielding the minimum distance (2:49). The critical problem in formulating a problem as a covering set problem is to set the distance measure (7:1309). In some problems, a direct distance is appropriate, while in others (such as those involving movement along streets) another distance measure must be devised. After the problem is formulated, an initial solution is chosen, and the algorithm follows the same pattern as the one-at-a-time exchange method. The check is for covering, though, instead of a cost function. This method does not guarantee an optimal solution, but has been shown to be very close to optimal in cases where the optimum is known (2:64).

Applicability of Location Analysis. The location analysis methods discussed here all assume that the demand is located at known points. The demand points in the AFSOF problem are the Army Special Forces targets, whose exact locations are highly classified. Therefore, loca-

tion analysis could not be used to provide an exact solution using the available data, but location analysis might be used to produce an approximate solution if some simplifying assumptions about the target locations are made. Rather than develop a new model to describe the AFSOF, though, this thesis effort will focus on modifying the current Hq MAC simulation model to incorporate multiple bases.

Simulation. Hq MAC is currently using a simulation model to answer questions concerning the AFSOF. This model, CRASOF, cannot be used in its present form to answer basing questions, however. As a simulation, though, it has been accepted by the headquarters staff as a good SOF model. This model can be modified to include multiple bases. With such a modification, if the new model is consistent with the old model, it would continue to be useful in dealing with other questions. Since this option is feasible and the modification can be made to conform with the available data, I will use simulation to model the AFSOF capability.

Methodology Overview

Theoretic Framework. The AFSOF system was modeled for a generic wartime scenario. The Army delivery requirements are input as mission rates with target locations modeled with probability distributions. Aircraft are assigned to various bases according to the basing strategy being tested. The system is modeled for a 90 day scenario.

Individual missions are planned to include any air refueling that may be necessary. Mission success is estimated using actual or projected aircraft mission capable rates, airborne mechanical abort rates, weather penetration capabilities, and attrition rates. If a mission is

unsuccessful, the mission request is reentered to be attempted the next day.

Measures of Effectiveness. Since the AFSOF cannot meet the US Army airlift requirements, a measure of the effectiveness of a given force under a certain basing strategy is the portion of the airlift requirement that can be met. This would be observed as the number of infil, exfil, and resupply missions actually accomplished over the simulation period. A secondary measure of effectiveness is the average delay between mission tasking and mission accomplishment. If the mission requirement rate is high at the beginning of a conflict, flying the missions toward the middle or end of the conflict would be ineffective. Therefore, timeliness must be considered in addition to actual numbers of missions flown.

Summary

The AFSOF is currently overtasked by the US Army, and the available funding is insufficient to close the gap between tasking and capability. Therefore, the effect of basing strategies on force capability must be considered. This study develops a simulation model and method to be used in making comparisons between different basing strategies. The final model and methodology will allow Hq MAC long range planners to add a quantitative element to choosing bases for Special Operations Forces. It will also be useful in force sizing exercises by long range planners. The model will not be designed for use in time critical situations. This chapter has stated the problem, and the next chapter will discuss the model developed to deal with the problem.

II. System Modeling

Introduction

The goal of the AFSOF is to provide airlift support to the US Army special forces. War planners have the task of deploying the AFSOF to meet the stated Army requirements. Since the current force is inadequate to meet the entire Army requirement, aircraft must be deployed so that they meet the maximum portion of the requirement possible. Therefore, there is a need for a model that can estimate the relative capabilities of a force under several basing strategies.

The model can be thought of as two interlocking models. One models the Army airlift requirements, while the other models Air Force activities. Army airlift requirements include target location; infiltration, resupply, and exfiltration rates; and A-team availability. Air Force activities include aircraft capabilities and availability, mission planning, and recycling aircraft and aircrews. The two models are not independent, since resupply and exfiltration rates depend on the infiltrations being accomplished. However, they will be described separately here.

This chapter will describe the assumptions made and logic used in both models. It will also cover the data required and the sources of that data.

US Army Special Forces Airlift Requirements

US Army special forces airlift requirements are stated in the form of target location distributions, infiltration rates, follow-up mission

delays, team recycle delays, and team availability. This section will describe each portion of the requirements and will include descriptions and sources of the data. It will then tie them together to show the logic of the model used.

Target Location. The actual US Army special forces target base is classified as Top Secret Specially Compartmented Information. Any model using the actual targets would have to be run at that classification, which would place a large restriction on its use. However, since the aim of this model is to provide insight and not to produce a detailed damage assessment, it does not need such a detailed data base. If the data base is aggregated, its classification is reduced to Secret, and it can be used on a much less restricted basis. In order to downgrade the classification of the target base, the theater of operations is broken into geographical regions and the percentage of total targets in that region is computed. The regions are also each assigned a regional priority and a threat probability distribution. Threat is broken into three categories: high, medium, and low. High threat is defined as requiring sophisticated threat evasion techniques and hardware, such as used by the MC-130. Low threat is defined as safe for aircraft such as the UH-1. Medium threat requires some threat avoidance, but not as much as high threat. Any area where the threat is higher than the high level will not have missions assigned, and so is not considered. The regions are described by their northwest and southeast corners, and the missions in each region are assumed to be evenly distributed throughout the region. The actual number of targets assigned in a run of the model is dictated by the infiltration rate used. An example target distribution

is shown below. This example is hypothetical and bears no resemblance to any real scenario.

Table I. Sample Target Distribution.

REGION	PRIORITY	PROB	NW CORNER		SE CORNER		THREAT HIGH	THREAT MED
1	4	.15	23N	105E	10N	110E	.20	.35
2	5	.10	20N	95E	10N	105E	.10	.65
3	2	.15	30N	110E	22N	120E	.65	.30
4	6	.10	33N	117E	30N	122E	.11	.73
5	3	.30	42N	115E	35N	125E	.24	.57
6	1	.20	44N	125E	38N	131E	.45	.45

The model assigns target locations as follows. When a mission request is generated, it is randomly assigned to one of the regions based on the probability distribution given. It is then assigned a location within the region based on the assumption that the target is equally likely to be at any point in the region. Along with a target location, the mission is randomly assigned a threat level based on the input threat distribution for the region. The mission request is also assigned a priority based on the regional priority and mission type. The regions are assumed to be homogeneous, so the actual features at a given location are not considered. This is a direct result of producing a secret data base from a top secret data base. The detail regarding target location was obscured by collecting large numbers of targets into regions. Also obscured by transforming the target data is the actual time sequencing of targets. There is provision in the model to input individual, preformatted missions, so the time sequencing problem can be partially alleviated. But using this feature could cause an increase in

the classification of a particular set of runs, so it should be used very carefully. The target data base can be obtained from Hq USAF/XOX, but it must be transformed into the above format.

Mission Rates. Mission rates include infiltration rates and resupply and exfiltration delays. The conflict is broken into as many as four phases, and an infiltration rate in terms of missions per day is set for each phase. A resupply delay is the number of days following a successful infiltration until the resupply request is to be entered into the system. An exfiltration delay is similar to a resupply delay. Also needed is the percentage of infiltrations which will not lead to exfiltrations. An example of the data needed is in Table II. As with the target location data, this data is purely hypothetical and bears no resemblance to any actual scenario.

Table II. Mission Rates.

Infil Rate:	5.0 per day for 10 days
	10.0 per day for 5 days
	5.0 per day thereafter
Resupply Delay:	5 days
Exfil Delay:	9 days
Percent w/o Exfil:	12.5

The mission request generator is based on the required infiltration rate. Mission requests are generated regardless of the availability of A-teams or aircraft. However, an infiltration mission request is not presented to the Air Force until a team is assigned. After a team is successfully inserted, the resupply and the exfiltration requests are generated as needed. Resupply requests are always generated, but exfil-

tration requests are not. The model provides the opportunity to insert guerrilla teams and resupply them indefinitely without exfiltration. Follow-up mission requests are generated after the appropriate delay. Resupply requests are generated again and again at the same time interval until the team is exfiltrated. The model also does not allow both a resupply and an exfiltration mission to be flown in support of the same team on the same day. The mission attempted first will be the only one attempted that day. The other mission will be delayed one day. When a team is exfiltrated, the model eliminates any outstanding mission requests for support of that team. All missions in support of a given team will be flown to the same location. This is a necessary simplification since the movements of teams on the ground are not modeled. It is a safe assumption, though, because missions in support of the same team will be close enough to each other to keep this from causing large errors. The required data can be acquired in the necessary form from Hq USAF/XOX.

A-team Availablility. Just as the Air Force has a limited number of aircraft assigned to a theater, the Army has a limited number of A-teams. The teams are divided into primary teams and reserve teams. The Army holds teams in reserve to take the place of attrited teams and to handle short notice missions. This model assumes that the only purpose of reserve teams is to replace primary teams that are lost and does not consider short notice missions. Since this model is a long range planning aid and short notice missions cannot be anticipated far in advance, this assumption is reasonable. When a team completes a mission, the Army has a planned recycle time before the team is assigned a new

mission. This recycle time is used for recuperation and mission preparation. Table III shows an example of the data needed. This data also bears no resemblance to real data.

Table III. A-team Availability.

Primary A-teams:	108 teams
Reserve A-teams:	12 teams
Recycle Delay:	7 days

When an infiltration request is generated, it is first presented to the Army to assign a team. If a team is available, it is assigned to the mission, and the request is sent to the Air Force to await aircraft. If a team is not available, the mission request waits for the next available team before asking for aircraft. When a team and aircraft are both assigned to the mission, the team is inserted. After a successful exfiltration, the team is released. After the required recycle delay, the team is then available for another infiltration. If the team is killed, it is replaced by a reserve team. If there are no reserve teams left, the number of primary teams is decreased. This data is also available in the necessary form from Hq USAF/XOX.

Summary. Army special forces airlift requirements are modeled by a target distribution, mission rates, and assigned teams. Infiltration requests are generated in accordance with the input mission rates. The infiltration requests wait for available teams. When a team is assigned, the request is then passed to the Air Force to await aircraft. A successful infiltration leads to a resupply request and, if necessary, an exfiltration request. If a mission is unsuccessful, it is resched-

uled for the next day. Only one mission to support a given team is attempted per day, and the mission not attempted is delayed one day. When a team is exfiltrated, any scheduled resupply mission requests are canceled. If a team is killed, a reserve team is moved to primary status to replace it. After a team is successfully exfiltrated, it is returned to available status after a delay for rest and mission preparation.

US Air Force Activities

US Air Force activities modeled here are aircraft capabilities, mission planning, aircraft and aircrew availability, recycling aircraft and aircrews, and asset location. The next section will describe how each of these activities is modeled and will tie them together in a description of the logic used.

Aircraft Capabilities. Aircraft capabilities are those characteristics used to distinguish one aircraft from another. They include: possible missions, threat penetration capability, cruise speed, maximum flying time without crew augmentation, unrefueled radius, air refueling capability, fuel capacity, fuel burn rate, mission effectiveness, mechanical abort rate, attrition rate, weather capabilities, crash survivability, and vertical takeoff and landing (VTOL) capability. The first two characteristics, possible missions and threat penetration capability, are used to assign an aircraft only to those missions it can accomplish. The next six, cruise speed through fuel burn rate, are used in planning missions. These are used to determine the mission plan to include any necessary air refueling and mission duration. The last six, mission effectiveness through VTOL capabilities, are used to determine

the actual outcome of individual missions. Table IV contains a sample of the aircraft capability data. The data shown is notional and does not reflect actual planning data. Actual data can be obtained from Hq MAC/XP and Hq MAC/DO.

When a mission request is presented to the Air Force, it has a mission type, target, and threat level associated with it. Mission type and threat level are the first two factors for eliminating possible aircraft assignments. If an infiltration request with a high threat level is presented, any aircraft to be considered for the mission must be able to both penetrate high threat levels and perform the infiltration. The model considers all aircraft for a given mission initially. After eliminating the impossible combinations, it then plans missions with all other aircraft. If an aircraft is a candidate and is at several bases, the base closest to the target is chosen. The mission is then planned, including any air refueling support required. Each combination is scored based on range, whether refueling is required, threat match, tanker availability, and mission priority. The range score gives preference to the aircraft that is closest to its maximum unrefueled range without exceeding it. This is to keep from using longer range aircraft to fly very short missions, which would make them unavailable to fly the longer missions. The refueling score prejudices the choice of aircraft in favor of an aircraft that does not need to refuel over an aircraft that does. This reduces the number of aircraft flying a particular mission, so it increases the chance of success. If an aircraft with high threat penetration capability is used for a low threat mission, it will be unavailable to fly a high threat mission, should one

Table IV. Aircraft Capabilities.

	MC-130	HH-53H	HC-130N	CV-22A
Prioritized Missions	Infil Resup Exfil	Exfil Infil Resup Rescue	Refuel Resup Infil	Exfil Infil Resup Rescue
Threat Capability	High	High	Med	High
Cruise Speed (KTAS)	220	120	220	220
Max Fly w/o Augment	9 HRS	10 HRS	9 HRS	9 HRS
Unrefueled Radius	950 NM	290 NM	1350 NM	575 NM
Air Refuelable	No	Yes	No	Yes
Fuel Capacity (LBS)	59000	11800	82000	16500
Mission Effectiveness	95%	95%	95%	95%
Mechanical Abort Rate	.43%	0%	2.44%	0%
Attrition Rate	.10%	.10%	.10%	.10%
Min Takeoff Ceil (FT)	0	0	0	0
Min Takeoff Vis (SM)	0.3	0.0	0.3	0.12
Infil Min Ceil (FT)	0	100	1000	0
Min Vis (SM)	0.0	0.25	3.0	0.0
Max Wind (KTS)	60	45	60	45
Rain Cancel	Yes	Yes	Yes	Yes
Turb Cancel	Yes	No	Yes	No
Prob Crew Surv Crash	15%	75%	15%	75%
VTOL Capability	No	Yes	No	Yes

arise. Therefore, the threat match score is added to reduce the chance of such a combination being chosen. Likewise, if a high threat capable tanker is used to refuel a low threat mission, it will be unavailable for higher threat missions. The tanker availability score is incorporated to reduce the chance of a tanker threat mismatch. The last scoring criterion is mission priority. Aircraft are best suited to their first priority missions, so an infiltration mission should be flown by an aircraft whose top priority mission is infiltration if possible. The mission priority score is designed to match aircraft to their top priority missions. After each eligible aircraft is scored for a mission, the one with the highest score is chosen to fly the mission. At this point, the last six aircraft characteristics are considered. Mission effectiveness is the probability that the mission will succeed given that the entire mission is flown. The mechanical abort rate is the probability that a mission is aborted due to mechanical failure and the weather abort rate is the probability that weather forces the aircraft to turn back. These three factors and the aircraft attrition rate have a direct effect on mission success. Mission effectiveness is modeled by taking a random draw after the mission is planned to decide if that particular mission is effective. One possible reason that a mission might be ineffective even though it reached the target area is that the drop zone might be compromised. Mechanical aborts and attrition are handled in the same manner. In this case, another random draw is made to locate the position of the abort or crash along the flight path.

Weather is dealt with differently. Random draws are made to determine takeoff and enroute weather for each aircraft involved with the

mission. Theater weather is modeled through probability distributions for ceiling, visibility, wind, rain, and turbulence. Each area is checked to see if the weather for a particular mission is good enough for the chosen aircraft to fly the mission. If any of the takeoff conditions are too bad, the mission is delayed. If any of the enroute conditions are too bad, the mission is aborted, and the position along the route for the abort is randomly chosen.

If an aircraft is lost, the assumption is that it crashed. At the point of the crash, a rescue mission is requested for the crew if the crew survives the crash. If the primary aircraft crashes or aborts, any support tankers return to their bases. In case of an abort, the mission is replanned with already committed aircraft. If a tanker crashes or aborts, the mission plan is recomputed starting with the first rendezvous missed. The only aircraft used are those already committed to the mission. If, as a result of a tanker missing a rendezvous, the primary aircraft is unable to return to base, the primary aircraft is counted as having crashed unless it is VTOL capable. In the latter case, it lands and is returned to the available force after an extra day delay. The assumption here is that the VTOL capable aircraft will find a place to set down until a tanker can arrive.

Individual missions are planned as follows. The mission is flown on a direct line between the aircraft's home base and the target. Refueling points are spaced along the mission track at regular intervals of less than twice the aircraft's unrefueled combat radius, and no refueling is planned within one combat radius of the objective. Tanker missions are planned from the tanker's home base to the first rendezvous

point for that tanker, along the primary aircraft's route to the tanker's last refueling, and back to the tanker's home base. If a mission is aborted, it is replanned starting at the point of deviation from the plan. The methods of choosing a base and planning a mission are much simpler than the detailed planning that goes into actual AFSOF missions. The actual mission planning requires detailed terrain and threat data bases. Bases are chosen based on the possible routes between them and the target locations and the terrain and threat along the routes. Routes are planned to use the terrain to mask the aircraft from the threat, so the routes are rarely direct. In order to plan missions realistically, the model would have to include the detailed terrain and threat data bases, and that is beyond the scope of this effort.

Aircraft and Aircrew Availability. Aircraft and aircrew availability is modeled through aircraft mission capable rates, aircrew ratios, expected aircraft turn times, minimum crew rest time, and mission preparation times. Mission capable rates are used through a random process, while the others are fixed rates and delays in the model. A notional sample of the data required follows in Table V. This data is not actual data.

Table V. *Aircraft and Aircrew Availability.*

	MC-130	HH-53H	HC-130N	CV-22A
Mission Capable Rate	61.5%	58.5%	64.5%	72.0%
Aircrew Ratio	1.50	1.50	1.50	2.00
Aircraft Turn Time (HRS)	1.50	2.00	1.50	1.00
Crew Rest (HRS)	12.0	12.0	12.0	12.0
SOF Mission Prep (HRS)	72.0	72.0	72.0	72.0
Rescue Mission Prep	12.0	12.0	12.0	12.0

At the beginning of each day, each available aircraft is checked with a random draw to determine if it is mission capable for that day. If it is not, it is not available until the following day. Aircrews are assigned by multiplying the number of aircraft initially available by the crew ratio for the particular aircraft. When a mission is flown, both the aircraft and the required aircrews are assigned to the mission. At the end of the mission, the aircraft are returned to available status after the minimum turn time delay. Aircrews are returned after crew rest and mission preparation time. The required data can be obtained from Hq MAC/DO and Hq MAC/LG.

Asset Location. This model includes aircraft basing information through a data base that includes the coordinates of the base and the numbers of aircraft and aircrews assigned. A sample data base is shown in Table VI. The data in it are notional and do not represent any actual scenario.

Table VI. Asset Location

		MC-130		HH-53H		HC-130		CV-22	
LAT	LON	ACFT	CREW	ACFT	CREW	ACFT	CREW	ACFT	CREW
14.8N	120.3E	3	5	0	0	5	7	5	10
26.5N	128.5E	2	3	5	7	0	0	0	0
32.8N	129.9E	3	5	0	0	5	8	3	6
35.9N	126.8E	0	0	7	12	0	0	2	4

When an aircraft is considered for a mission, its location is of primary importance. If several bases have that type aircraft, the base closest to the target is chosen to supply the aircraft and aircrews needed. When a mission is completed, the aircraft and aircrews are

returned to their base of origin. The model keeps track of the bases of origin for each aircraft involved in the mission. The necessary data can be obtained from Hq MAC/DO and Hq MAC/XO.

Summary

The model used here can be decomposed into two interlocking models. One models the US Army requirements, and the other models the AFSOF's activities. Army requirements are based on number of teams, number and location of targets, mission rates, and inter-mission delays. The data for describing the Army requirements is available from Hq USAF/XO. However, the target location data must be aggregated to reduce its classification. Air Force activities are based on aircraft capabilities, mission planning, aircraft and aircrew availability, and recycling delays. The data for describing the Air Force capabilities is available from Hq MAC. Chapter III will detail the actual implementation of the model described in this chapter.

III. Model Coding and Testing

Introduction

Chapter II identifies the logic used in the model. This chapter covers the coding of the model in the Simulation Language for Alternative Modeling (SLAM) and FORTRAN for use on the VAX 11/785 Classroom Support Computer (CSC) at the Air Force Institute of Technology. Included is a history of the development of the original CRASOF model at Hq MAC. A more detailed explanation of the model coding is in Appendix A. The SLAM code appears in Appendix B, and the FORTRAN code can be obtained from Hq MAC/XP-CAAG, Scott AFB, IL 62225.

History

In 1983, Hq MAC/XPQ tasked Hq MAC/XPS and 23AF/XP to conduct analyses to determine the requirement for HC-130 tanker aircraft for the support of Combat Rescue (CR) and Special Operations Forces. In late 1983, Maj Jack Dickinson began the Hq MAC study, and I was tasked by 23AF/XP to work with him on the study. The initial study of the problem led Maj Dickinson to conclude that a simulation model was necessary to ensure the decision makers would accept any recommendation. Since the study was to answer a question about the need for tanker support, particular attention was paid to the air refueling logic used in the model. The target location data was available only in the form of a distance distribution, so the model assumed collocation of all assets. The target range distributions were computed using current bases and known target areas. The distance to each target was computed from the most likely base to support the mission to that target along the expected

route. If the bases were changed, the range distributions would have to be changed as well. Maj Dickinson and I spent all of 1984 developing the CRASOF model with frequent changes needed to capture the refueling logic used by planners. We designed the model to be as flexible as possible to allow its use in answering other questions. In the course of working on the tanker study, other offices in Hq MAC and some offices in Hq USAF raised questions about force sizing, the impact of increased weather capabilities, and the impact on the AFSOF of the introduction of new weapon systems such as the HH-60 and the CV-22A. In early 1984, we used the CRASOF model to build the CR and AFSOF Minimum Risk Forces for Hq MAC/XPP. The tanker study was completed in mid-1984 after I had left 23AF. As a result of these two uses of the CRASOF model, it has been accepted within Hq MAC as an analytical tool to be used in describing CR and AFSOF. This thesis effort has extended the original CRASOF model to include a multi-basing capability. The limitation imposed by the restricted target data has been lifted since Hq USAF/XOX identified the need for more detailed information from the US Army. The improved CRASOF model is intended for use by Hq MAC in place of the original CRASOF model.

Modifications and Additions

The modifications and additions were all associated with expanding the model to include geographical locations of both aircraft bases and targets. Modifications were made to the target location input data base, the target choice logic, mission planning logic, and the aircraft and aircrew allocation and recycling logic. An aircraft basing data

base was added along with the capability to compute distances between points on the earth.

Target Location. The original model target data base was a probability distribution of distances from the aircraft base. Target location was defined as a distance, not a location. In the modified model, the target input data base was changed to include geographical regions, as is shown in Table I in the last chapter, and the target choice logic was changed to correspond with the change in data base. Before, the target distance was chosen, but in the new model a location is chosen. The logic used is explained in Chapter II.

Mission Planning. The original CRASOF model used only distances to locate points, but the new model uses geographical coordinates. Therefore, the mission planning has been modified to calculate the coordinates of all refueling points, abort points, and crash locations. It also calculates the support aircraft data based on the support base, which may differ from the primary aircraft's base.

Aircraft and Aircrew Allocation. In the original model, all assets were located at the same base. Since that assumption has been deleted, logic was added to the allocation and recycling subroutines to track not only the aircraft and aircrews, but also the bases they were assigned to.

Aircraft Basing Data Base. Since it is no longer assumed that all aircraft are collocated, an additional input data base regarding the various bases was added. An example data base is shown in Table VI. The data base includes base location, aircraft assigned, and crews assigned. When an aircraft from a given base is assigned to a mission,

the number of that type aircraft at that base is decremented to indicate that the aircraft is no longer available. The same logic is applied to aircrews.

Distance Computation. The original model had no need for distance computations, so the subroutine Distance and the function Arccos were added to calculate the distance between two points given their coordinates.

Model Verification

The original CRASOF model was verified at Hq MAC in late 1984 and early 1985. Therefore, the new CRASOF model need not be verified entirely, but only in those portions that were modified or added and in portions where the changes might impact the other code. Verification of the rest of the model is accomplished by ensuring the code is unchanged from the original model.

Target Location. The target location changes were tested first by including diagnostic print statements in the FORTRAN code to check that the code implemented the desired logic. This included checking to see if the target coordinates were within the region chosen. The code did implement the logic properly. Then the model was run with statistics taken on the regions chosen for the missions. The actual output distribution of regions was compared with the input probability distribution and tested to see if the output was consistent with the input. The tests showed that the the input and output were consistent. The new target location code completely replaced the old code but did not affect any other portion of the model. These tests were considered adequate to verify that the new target location logic was implemented properly.

Mission Planning. The new mission planning logic was more complicated than the new target location logic and did not completely replace the old mission planning logic. However, the same steps were used to verify the code. The model was run with the diagnostic print statements for ten days rather than a complete 90 day scenario because the output was too large for any longer period. The code correctly implemented the desired mission planning logic. The results of this run were compared with results from the old logic for consistency. The results were consistent, and the mission planning code was determined to be correct.

Aircraft and Aircrew Allocation. The aircraft and aircrew allocation logic in the original model were separate because of the assumption of collocation of assets. They were combined in the new model. Diagnostic print statements were embedded in the allocation code, and the code was tested using a short run in the same manner as the mission planning code. The code correctly implemented the desired logic. The results were compared with results from the original model, and were found to be consistent.

Aircraft Basing Data Base. The basing data base was checked during both the input routine and during the aircraft and aircrew allocation verification check. The data base was read in correctly. The allocation code properly assigned assets to missions and returned the assets to the correct bases. The aircraft basing data base was properly integrated into the model.

Distance Computation. The distance computation routines were tested twice. First, they were checked outside the model to ensure they accurately converted location data into distances. Then they were

tested in the model to ensure they were called correctly and that their output was correct given the calling arguments. The routines worked correctly in both tests.

Interactions. An important part of the model verification was ensuring the changes and modifications did not change the rest of the model. This was verified by checking the output with the diagnostics included to see that the logic followed was unchanged in the unmodified portions of the model. In all cases, the modifications did not change the rest of the model.

Summary. Model verification was done by first checking the code to ensure it properly implemented the desired logic. Then the model was run with diagnostic print statements embedded in the code to trace the logic used and to show the intermediate results. These embedded diagnostics verified that the encoded model accurately implements the logical model.

Model Validation

The original CRASOF model is the only available standard by which to judge the new model. There is no operational test data or exercise data to use. However, since the original model used only a single base, it is not an accurate measure for validation of a multibase model. In model verification, the intermediate results of the new model were found to be consistent with the intermediate results of the old model, but the final results cannot be compared because of the difference in basing assumptions. Therefore, model validity can be based only on the validity of the assumptions made and logic used in the model. The assump-

tions made are the same as in the original model except for two: 1) aircraft are distributed at different bases, and 2) targets are geographically located. The assumptions in the original model were accepted by the MAC staff, so they are considered valid. The two new assumptions have been coordinated with Hq MAC/XPCS, so they are considered valid. The logic used in the new model has also been coordinated with Hq MAC/XPCS. Therefore, since the assumptions and logic are accepted and the logic is correctly implemented by the model, the new CRASOF model is assumed to be valid.

Summary

The model developed in this thesis is a modification of a current Hq MAC/XPCS model, CRASOF. The modifications to the original model are changes in the target location data base and the target choice logic, changes in the mission planning logic, and changes in the aircraft and aircrew allocation and recycling logic. Additions to the model are an aircraft basing data base and distance computation logic. These modifications and additions were individually and collectively tested to verify proper implementation of the model logic. The assumptions made and logic used were coordinated with Hq MAC/XPCS to validate the model (4). The resulting model is an improvement over the original CRASOF model because it permits assessment of different basing strategies. The following chapters describe the use of this model to choose among different basing strategies.

IV. Scenario and Experimental Design

Scenario

The following scenario is not an actual scenario and does not closely resemble any scenario currently considered by Hq MAC (4).

In the year 2001, Communist China has a change of foreign policy due to a turnover in government. The new foreign policy is aggressive and leads China to attack Southeast Asia (SEA) in an effort to take over the peninsula. China succeeds in its efforts in SEA and begins to build up forces in North Korea. The United States enters the conflict to stop any further expansion. Thirty days before actually declaring war, the US commits its Army and Air Forces Special Operations Forces, and the SOF involvement continues for sixty days past a formal declaration of war. South Korea, Japan, and the Philippines are allied with the US, and authorize basing privileges for use by the SOF. The SOF target areas and mission distributions are shown in Table VII.

Table VII. Target Distribution.

REGION	PRIORITY	DESCRIPTION	TARGET PROB
1	3	Republic of Vietnam	.15
2	6	Kampuchea and Thailand	.10
3	5	China south of Shanghai	.15
4	2	Shanghai and the Hangchoy Bay	.10
5	1	Bei-jing and the Gulf of Chihli	.30
6	4	North Korea	.20

US Army Assets. One hundred twenty US Army A-teams are available. The mission rates and team availability rates are shown in Table VIII.

Table VIII. Mission and A-team Availability Rates.

Infil Rate:	5.0 per day for 10 days
	10.0 per day for 5 days
	5.0 per day thereafter
Delay Resupply:	5 days
Exfil:	9 days
Recycle:	7 Days
Percent w/o Exfil:	12.5
A-teams Primary:	108 teams
Reserve:	12 teams

US Air Force Assets. The USAF aircraft and aircrews that are available in this scenario are shown in Table IX. Aircraft characteristics are shown in the input file SOFAC in Appendix C.

Table IX. USAF Assets Available.

Aircraft	Number	Crews
MC-130	7	11
HH-53H	7	11
HC-130 (II)	3	5
HC-130 (I)	3	5
CV-22A	7	14

Factors of Interest.

Two factors are of interest in this problem. The first is the basing strategy used, and the second is weather. In this demonstration, three basing strategies and two levels of weather were used. Basing strategy is the main factor in the analysis, and weather was chosen because it plays an important part in estimating force capability. The basing options considered are discussed in the next section, and weather

is discussed in the section following that.

Basing Options. One feature of the current SOF is that it has few aircraft, as is shown in Table IX. Three alternate means of deploying these limited assets were developed. The three options are shown in Table X. The four bases considered are Clark AB, Philippines; Okinawa; Nagasaki, Japan; and Kunsan AB, South Korea. Option 3 is the closest basing strategy to anticipated peacetime basing. This option would require the fewest changes in preparing for the clandestine SOF missions. Option 1 is designed to spread out the tanker support capability to allow more flexibility for the exfiltration missions. It also moves the CV-22A closer to the priority 1 region, which should reduce refueling requirements. Option 2 spreads the tanker force further. It also reshuffles the vertical lift assets, the HH-53H and the CV-22A, to see if this adds capability.

Weather. Weather can play an important part in force capability. The two weather categories considered in this analysis were summer and winter. The actual weather data used is shown in Appendix C in the input file SOFWX. Summer and winter were chosen because they represent the two extremes in expected weather. One basing strategy may result in very different force capabilities under the two extremes. If one strategy exhibits such varying results and another is less sensitive to weather, the second strategy may be preferable. This is because war plans do not vary based on the seasons, but are constant regardless of the weather.

Experimental Design

A full factorial design was used because there were only two fac-

Table X. Basing Options.

Option 1

ACFT	CLARK NBR	CLARK CRWS	OKINAWA NBR	OKINAWA CRWS	NAGASAKI NBR	NAGASAKI CRWS	KUNSAN NBR	KUNSAN CRWS
MC-130	4	6	3	5	-	-	-	-
HH-53H	-	-	-	-	-	-	7	11
HC-130II	1	2	2	3	-	-	-	-
HC-130I	1	2	2	3	-	-	-	-
CV-22A	-	-	3	6	4	8	-	-

Option 2

ACFT	CLARK NBR	CLARK CRWS	OKINAWA NBR	OKINAWA CRWS	NAGASAKI NBR	NAGASAKI CRWS	KUNSAN NBR	KUNSAN CRWS
MC-130	4	6	3	5	-	-	-	-
HH-53H	-	-	3	5	-	-	4	6
HC-130II	1	1	1	2	1	2	-	-
HC-130I	1	2	1	2	1	1	-	-
CV-22A	3	6	-	-	2	4	2	4

Option 3

ACFT	CLARK NBR	CLARK CRWS	OKINAWA NBR	OKINAWA CRWS	NAGASAKI NBR	NAGASAKI CRWS	KUNSAN NBR	KUNSAN CRWS
MC-130	4	6	3	5	-	-	-	-
HH-53H	-	-	-	-	-	-	7	11
HC-130II	3	5	-	-	-	-	-	-
HC-130I	3	5	-	-	-	-	-	-
CV-22A	3	6	4	8	-	-	-	-

tors and six combinations to consider. The effects to be considered are the direct effects of the basing options and weather and the interaction effect of basing with weather.

Sample Size

Two basing options are considered to be equivalent in mission capability in this exercise if the difference in missions accomplished

over the ninety day period is less than ten missions in each mission category and if the difference in average mission delay is less than twelve hours in each mission category. Therefore, the sample size must be large enough to detect such a difference. Pilot runs were made to get an estimate of the variance in the measured performance. Ten runs were determined to be sufficient to detect the established critical differences in all categories.

Variance Reduction Techniques.

Variance reduction techniques were not applied in this analysis because of the size of the model and the many uses of the random streams. Only ten random streams are available in SLAM, and there are more than twice this number of uses of the streams in the model. Every attempt was made to keep the streams synchronized to reduce variance, but absolute synchronization was not verifiable in a reasonable amount of time. Therefore, rather than spend extra time trying to verify stream synchronization between runs, only sample size was considered as a means of reducing the variance.

Summary

The scenario chosen to demonstrate the use of the model is a war against the People's Republic of China. This scenario is not a scenario used by war planners. Four potential bases were selected: Clark AB, Okinawa, Nagasaki, and Kunsan AB. Two factors were chosen to vary: basing option and weather. Three different basing options were chosen using the four bases. Weather was varied between summer and winter. A full factorial design was used for the experiment, so there were six

treatments. Ten replications were made for each treatment to ensure the experiment would identify differences of ten successful missions in each mission category and delay differences of twelve hours or more. No variance reduction techniques were used, although every effort was made to synchronize the use of the random streams. Chapter V will present the results of the experiment and will also include some excursions.

V. Analysis of Results

Introduction

This chapter will first discuss the statistical results of the experiment and will then interpret these statistical results to draw real world conclusions. The chapter will also include several excursions that were run on the effect of changing the relative priorities of infiltration and exfiltration missions. Finally, an example of how the model can be used in a quick response study will be described.

Technical Description

The full factorial design described in Chapter IV was run for ten replications for each treatment level. The output data was analyzed using the SAS statistical software package (5). The response variables of interest were total successful missions, successful infiltrations, successful exfiltrations, successful resupplies, average infiltration delay, average exfiltration delay, and average resupply delay. An analysis of variance (ANOVA) was conducted to test for differences in treatment means in each category. Duncan's multiple range test was used to conduct simultaneous pairwise comparisons of the treatment means. The factors were basing option (OPT), weather (WX), and the interaction of OPT and WX. The ANOVA tables are shown in Tables XI through XVII.

Comparison of Treatment Means. The seven ANOVA tables below show that weather has no significant impact on any of the response variables. The basing option chosen does have a significant impact on all response variables except the number of successful exfiltrations accomplished. Therefore, the effect of weather on the mean response was not checked.

Table XI. ANOVA for Successful Missions.

DEPENDENT VARIABLE: SUCC

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	22761.283	4552.257
ERROR	54	84631.300	1567.246
CORRECTED TOTAL	59	107392.583	

MODEL F = 2.90 PR > F = 0.022

R-SQUARE	C.V.	ROOT MSE	SUCC MEAN
0.212	6.104	39.588	648.583

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	22219.633	7.09	0.002
WX	1	28.017	0.02	0.894
OPT*WX	2	513.633	0.16	0.849

Table XII. ANOVA for Successful Infiltrations.

DEPENDENT VARIABLE: SINF

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	1209.283	241.857
ERROR	54	8123.300	150.431
CORRECTED TOTAL	59	9332.583	

MODEL F = 1.61 PR > F = 0.174

R-SQUARE	C.V.	ROOT MSE	SINF MEAN
0.130	6.069	12.265	202.083

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	1124.433	3.74	0.030
WX	1	12.150	0.08	0.777
OPT*WX	2	72.700	0.24	0.786

Table XIII. ANOVA for Successful Exfiltrations.

DEPENDENT VARIABLE: SEXF

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	220.083	44.017
ERROR	54	6634.100	122.854
CORRECTED TOTAL	59	6854.183	
MODEL F =	0.36		PR > F = 0.875

R-SQUARE	C.V.	ROOT MSE	SEXF MEAN
0.032	7.010	11.084	158.117

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	144.633	0.59	0.559
WX	1	2.017	0.02	0.899
OPT*WX	2	73.433	0.30	0.743

Table XIV. ANOVA for Successful Resupplies.

DEPENDENT VARIABLE: SRES

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	20648.200	4129.640
ERROR	54	49858.200	923.300
CORRECTED TOTAL	59	70506.400	
MODEL F =	4.47		PR > F = 0.002

R-SQUARE	C.V.	ROOT MSE	SRES MEAN
0.293	10.536	30.386	288.400

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	20560.900	11.13	0.000
WX	1	9.600	0.01	0.919
OPT*WX	2	77.700	0.04	0.959

Table XV. ANOVA for Infiltration Delay.

DEPENDENT VARIABLE: INDL

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	29.579	5.916
ERROR	54	59.891	1.109
CORRECTED TOTAL	59	89.470	

MODEL F = 5.33 PR > F = 0.001

R-SQUARE	C.V.	ROOT MSE	INDL MEAN
0.331	7.584	1.053	13.409

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	29.549	13.32	0.000
WX	1	0.008	0.01	0.933
OPT*WX	2	0.022	0.01	0.990

Table XVI. ANOVA for Exfiltration Delay.

DEPENDENT VARIABLE: EXDL

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	17.084	3.417
ERROR	54	32.847	0.608
CORRECTED TOTAL	59	49.932	

MODEL F = 5.62 PR > F = 0.001

R-SQUARE	C.V.	ROOT MSE	EXDL MEAN
0.342	34.674	0.780	2.249

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	16.834	13.84	0.000
WX	1	0.150	0.25	0.622
OPT*WX	2	0.100	0.08	0.921

Table XVII. ANOVA for Resupply Delay.

DEPENDENT VARIABLE: RSDL

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
MODEL	5	2.746	0.549
ERROR	54	4.207	0.078
CORRECTED TOTAL	59	6.953	

MODEL F = 7.05 PR > F = 0.000

R-SQUARE	C.V.	ROOT MSE	RSDL MEAN
0.395	23.408	0.279	1.192

SOURCE	DF	TYPE I SS	F VALUE	PR > F
OPT	2	2.746	17.62	0.000
WX	1	0.000	0.00	0.956
OPT*WX	2	0.000	0.00	0.996

Duncan's multiple range test was used to compare treatment means between the different basing options with alpha=0.10. The results are in Tables XVIII and XIX. Table XX contains a ranking of the three basing options for all seven response variables. Responses for which there is no significant difference at the specified alpha level between two options are noted by equal rankings for those two options.

Model Aptness. All of the results shown so far are based on two assumptions for each response variable. The first is that the observed responses are normally distributed. The second is that the variance of the observations is constant across all treatment levels. These assumptions can be tested by checking the residuals. Figures 1 through 7 contain the normal probability plots for all response variables and Figures 8 through 14 contain plots of the residuals versus the predicted

Table XVIII. Missions Accomplished.

	OPT 1	OPT 2	OPT 3
TOTAL (*)	673	645	628
INFILTRATION (**)	202	207	197
EXFILTRATION (+)	157	160	158
RESUPPLY (++)	314	278	273

(*) - OPTION 1 IS SIGNIFICANTLY GREATER THAN OPTION 2 OR 3. OPTIONS 2 AND 3 DO NOT DIFFER SIGNIFICANTLY.

(**) - ONLY SIGNIFICANT DIFFERENCE IS BETWEEN OPTIONS 2 AND 3.

(+) - NO SIGNIFICANT DIFFERENCES

(++) - OPTION 1 IS SIGNIFICANTLY GREATER THAN OPTION 2 OR 3. OPTIONS 2 AND 3 DO NOT DIFFER SIGNIFICANTLY.

Table XIX. Mission Delay (Days).

	OPT 1	OPT 2	OPT 3
INFILTRATION (*)	12.42	13.93	13.88
EXFILTRATION	2.23	1.61	2.91
RESUPPLY	1.17	1.47	.94

(*) - NO SIGNIFICANT DIFFERENCE BETWEEN INFILTRATION DELAYS FOR OPTIONS 2 AND 3. ALL OTHER DIFFERENCES ARE SIGNIFICANT.

Table XX. Ranking of Basing Options.

RESPONSE	OPTION 1	OPTION 2	OPTION 3
TOTAL MISSIONS	1	2	2
INFILTRATIONS	1/2	1	2
EXFILTRATIONS	1	1	1
RESUPPLIES	1	2	2
INFILTRATION DELAY	1	2	2
EXFILTRATION DELAY	2	1	3
RESUPPLY DELAY	2	3	1

responses. In all the plots, A indicates one observation, B indicates two, etc. The normal probability plots all show that the residuals are approximately normally distributed. The residual vs. predicted plots show no serious differences in variance. Therefore, the data does not deviate from the assumptions of normality and equal variances sufficiently to invalidate the conclusions drawn from the ANOVA.

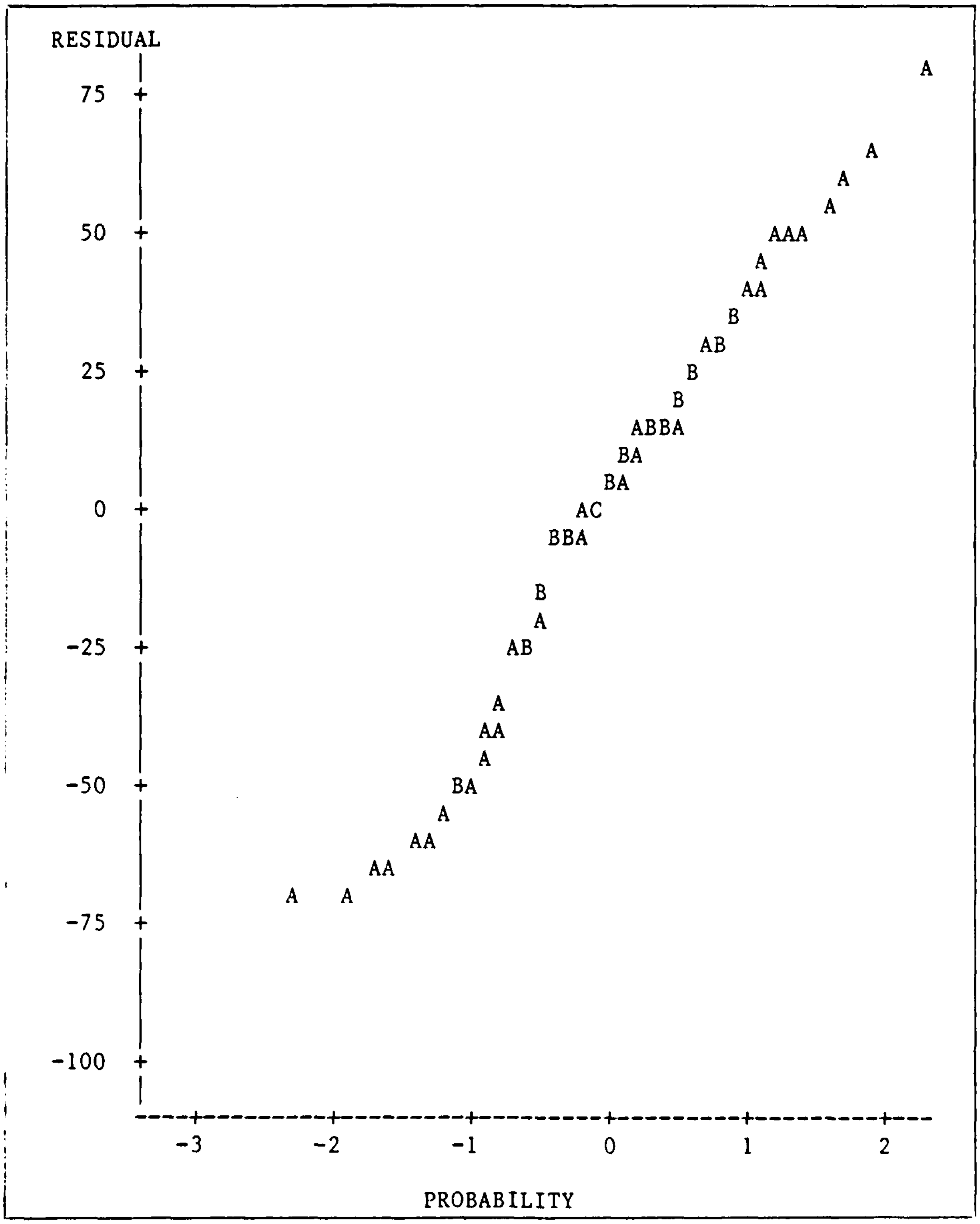


FIGURE 1.
NORMAL PROBABILITY PLOT FOR TOTAL SUCCESSFUL MISSIONS.

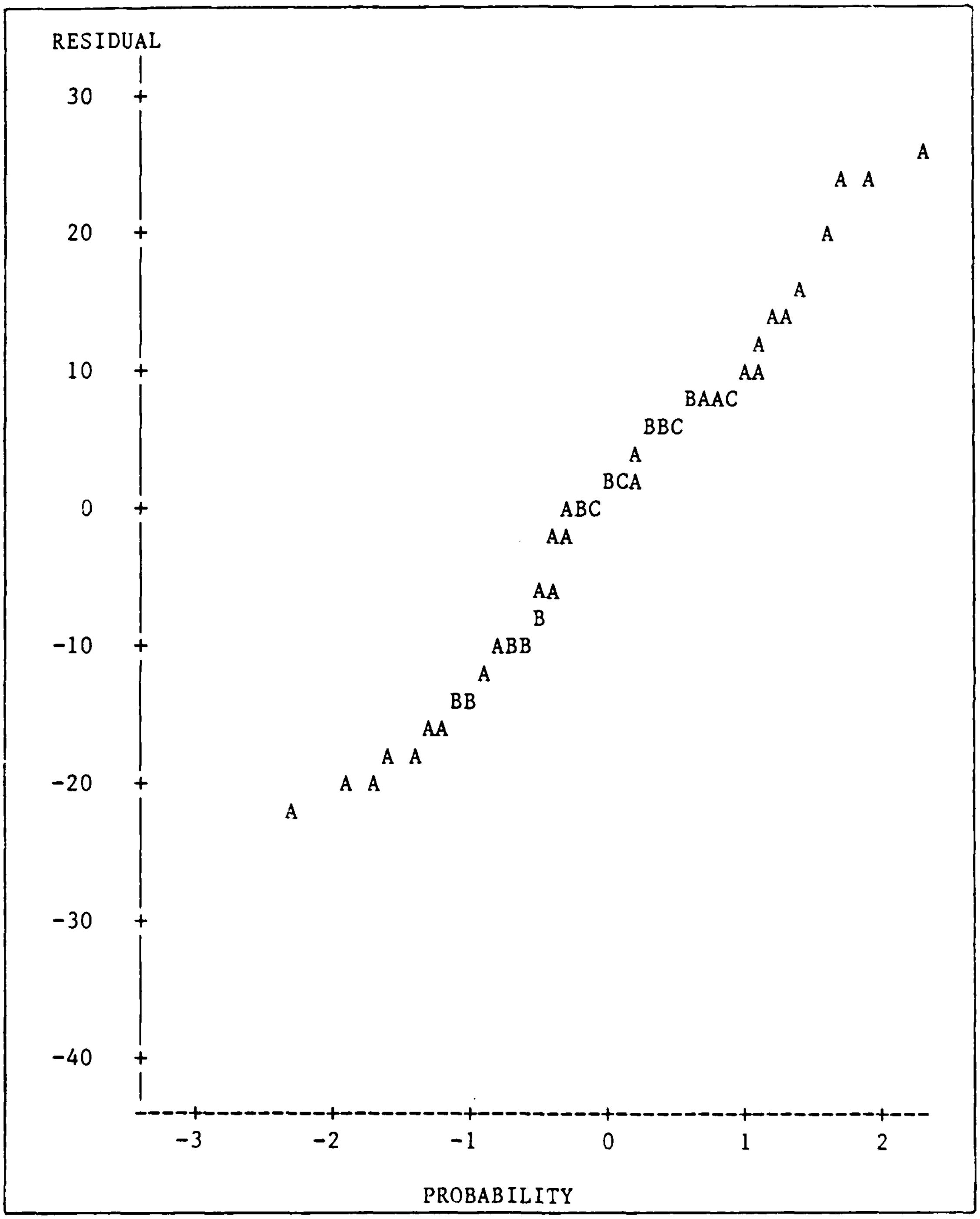


FIGURE 2.
NORMAL PROBABILITY PLOT FOR SUCCESSFUL INFILTRATIONS.

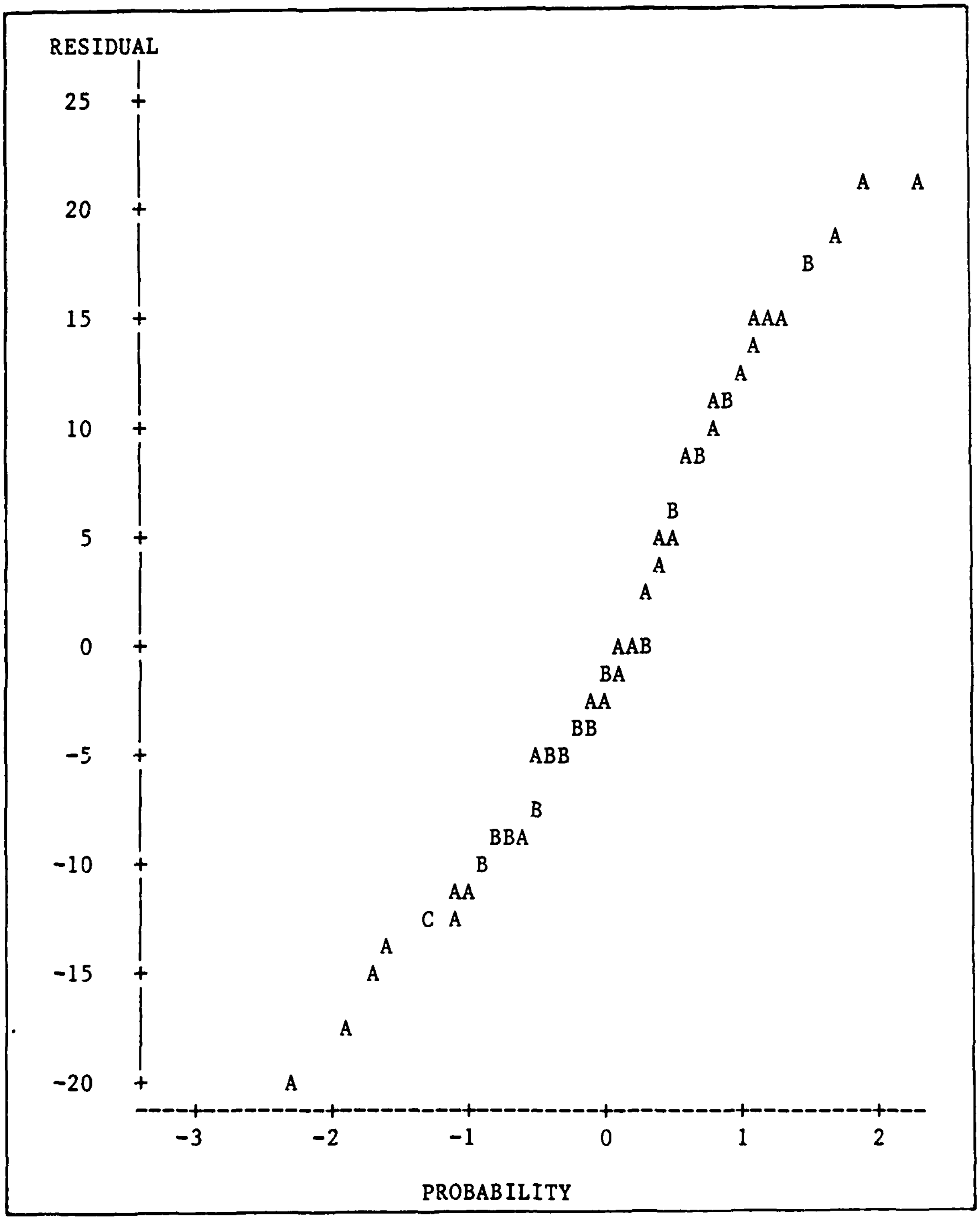


FIGURE 3.
NORMAL PROBABILITY PLOT FOR SUCCESSFUL EXFILTRATIONS.

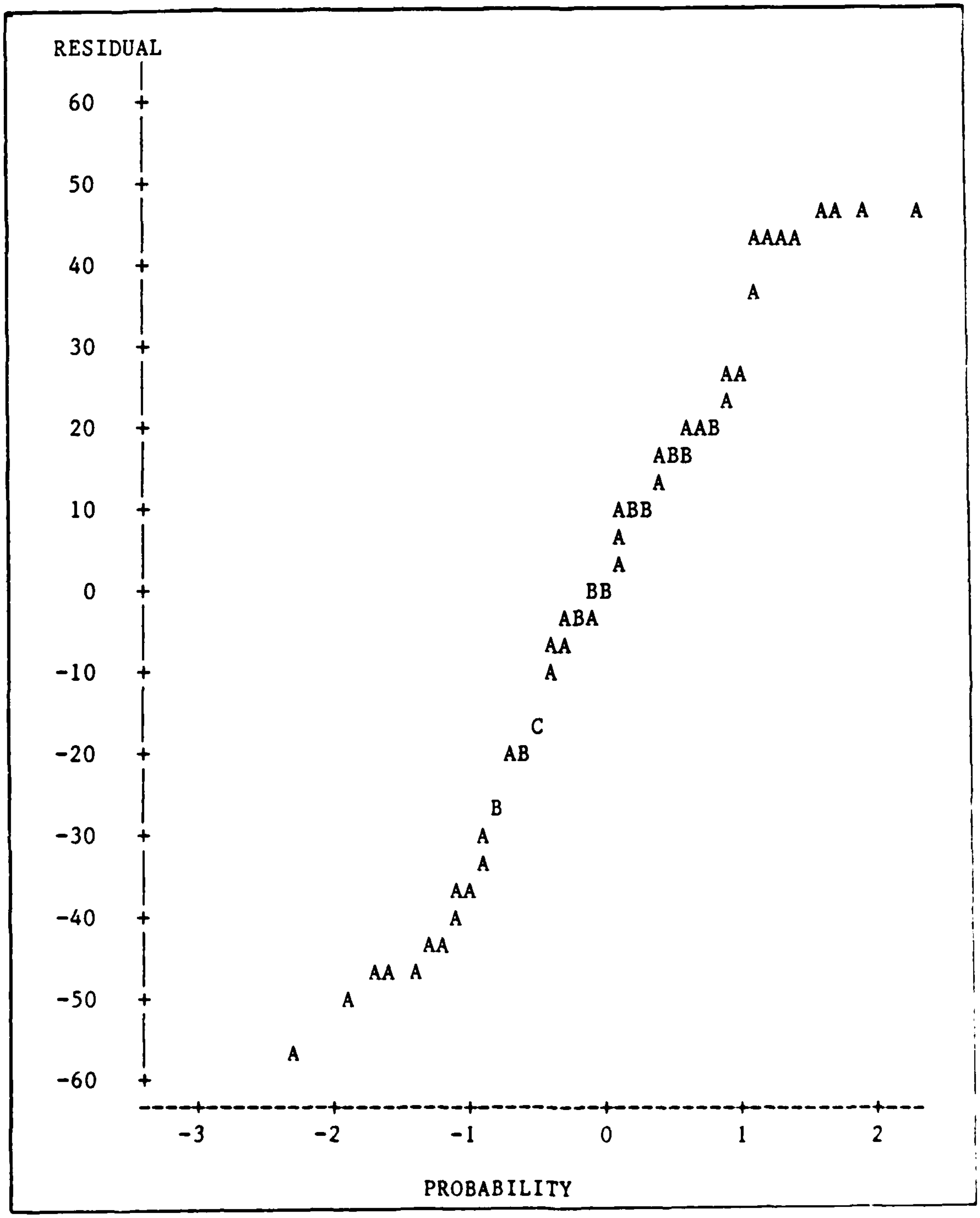


FIGURE 4.
NORMAL PROBABILITY PLOT FOR SUCCESSFUL RESUPPLIES.

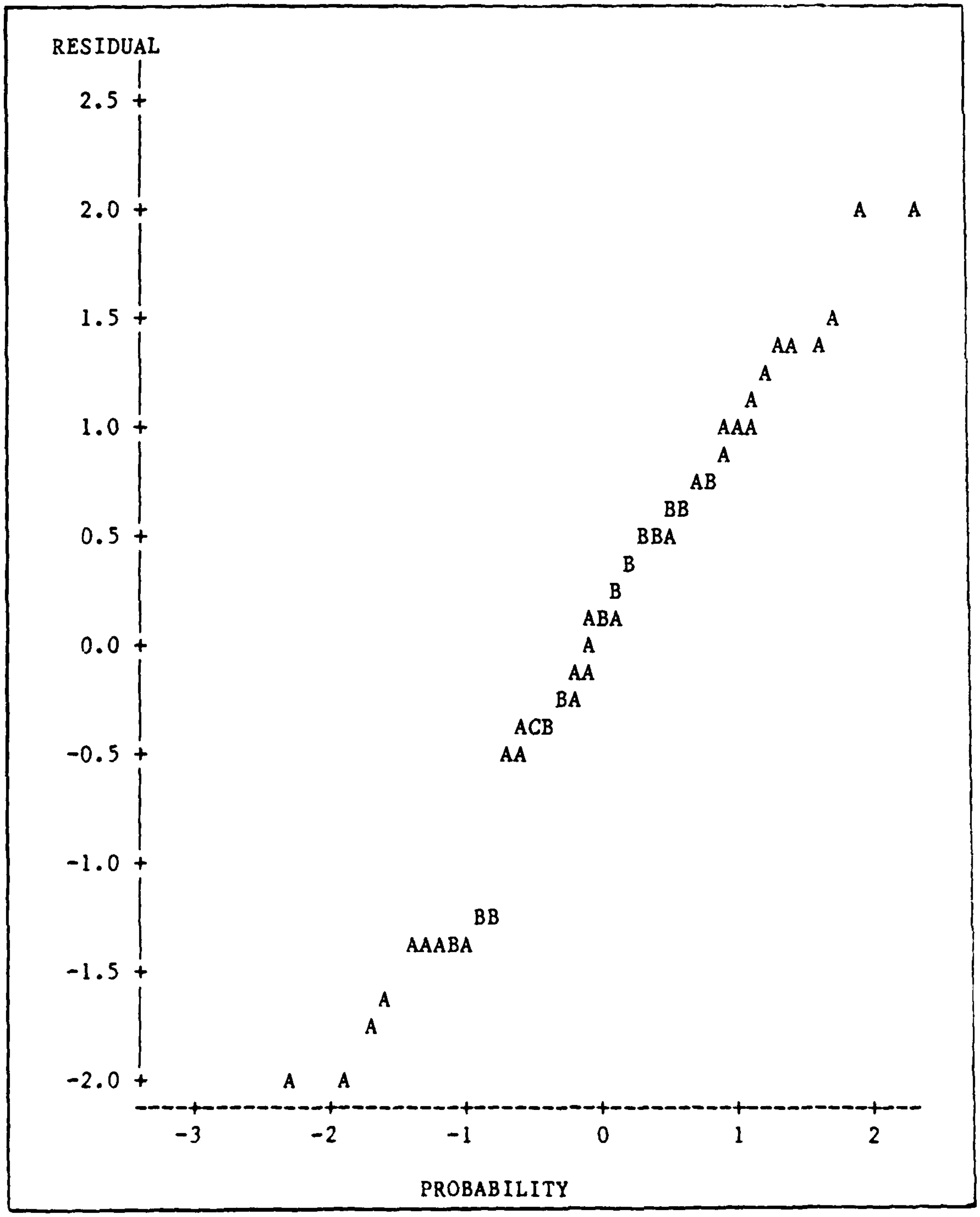


FIGURE 5.
NORMAL PROBABILITY PLOT FOR INFILTRATION DELAY.

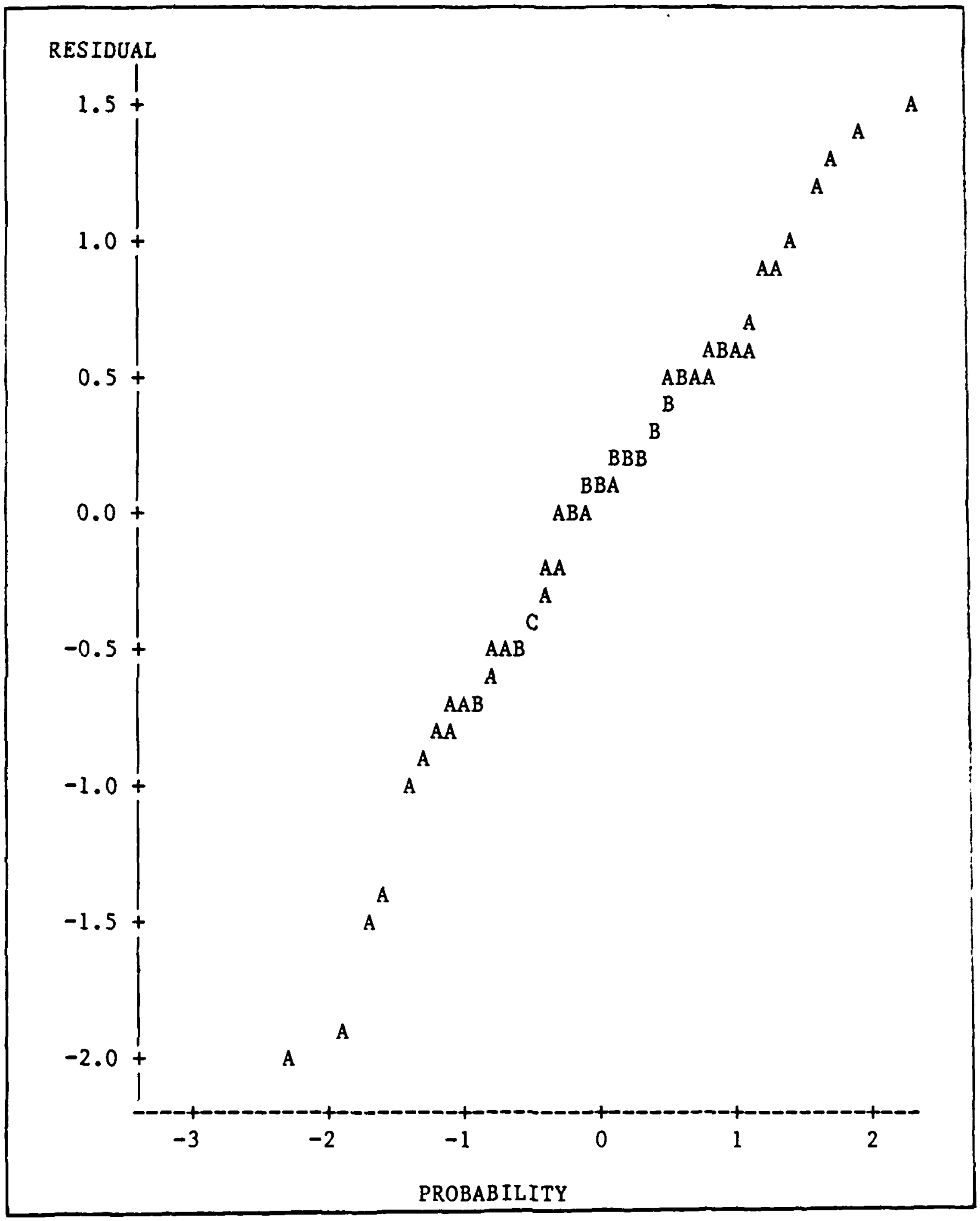


FIGURE 6.
NORMAL PROBABILITY PLOT FOR EXFILTRATION DELAY.

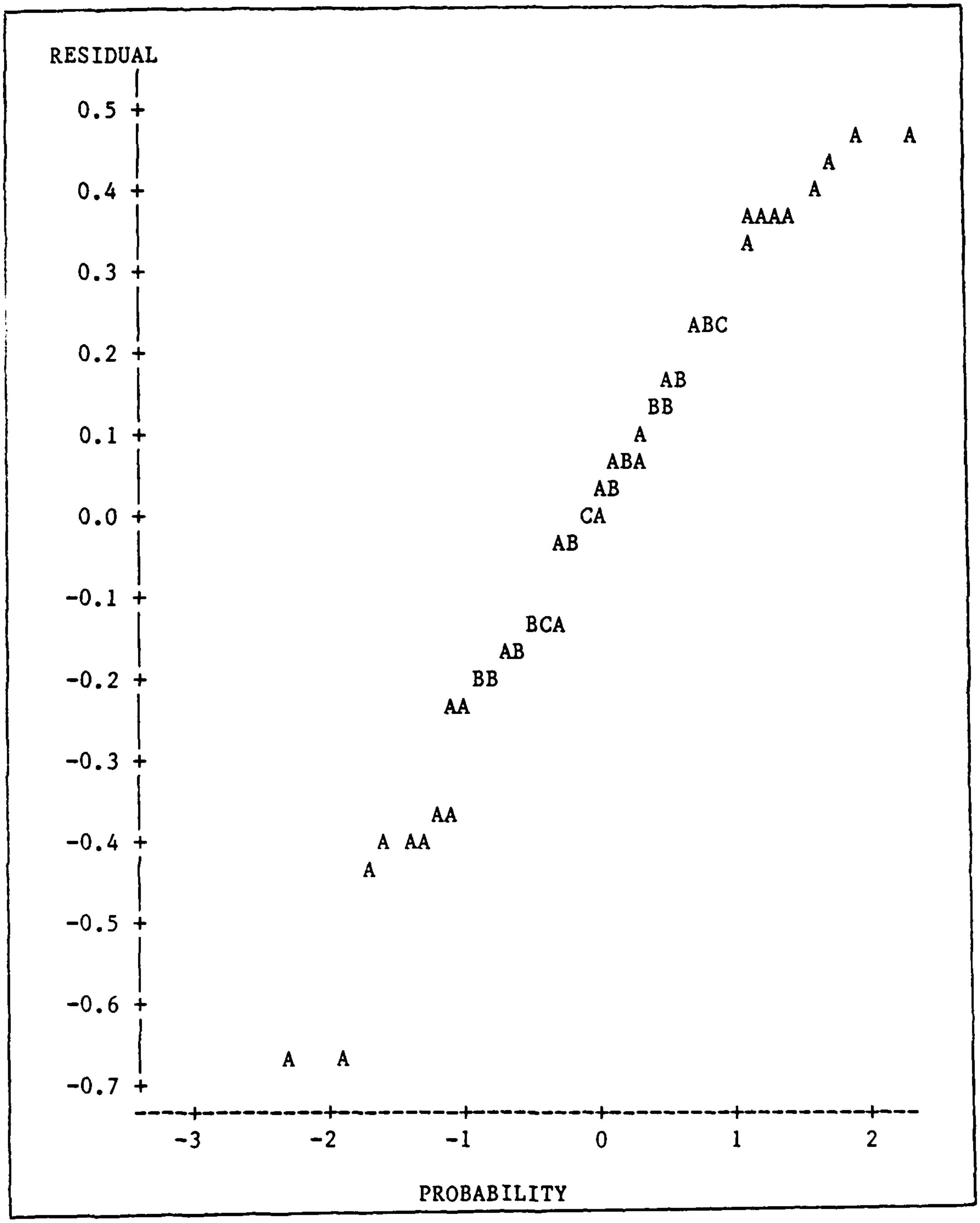


FIGURE 7.
NORMAL PROBABILITY PLOT FOR RESUPPLY DELAY.

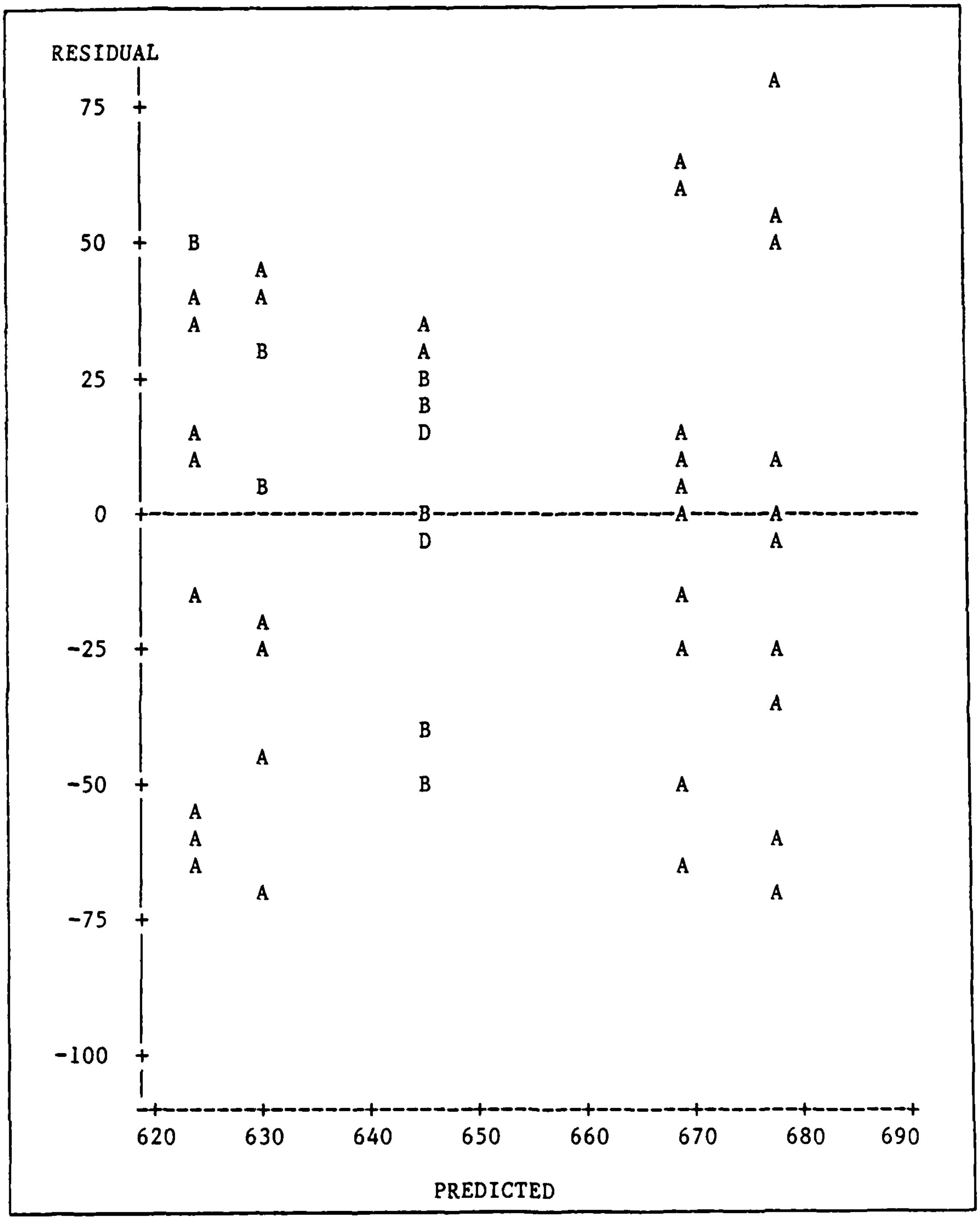


FIGURE 8. RESIDUALS FOR TOTAL SUCCESSFUL MISSIONS.

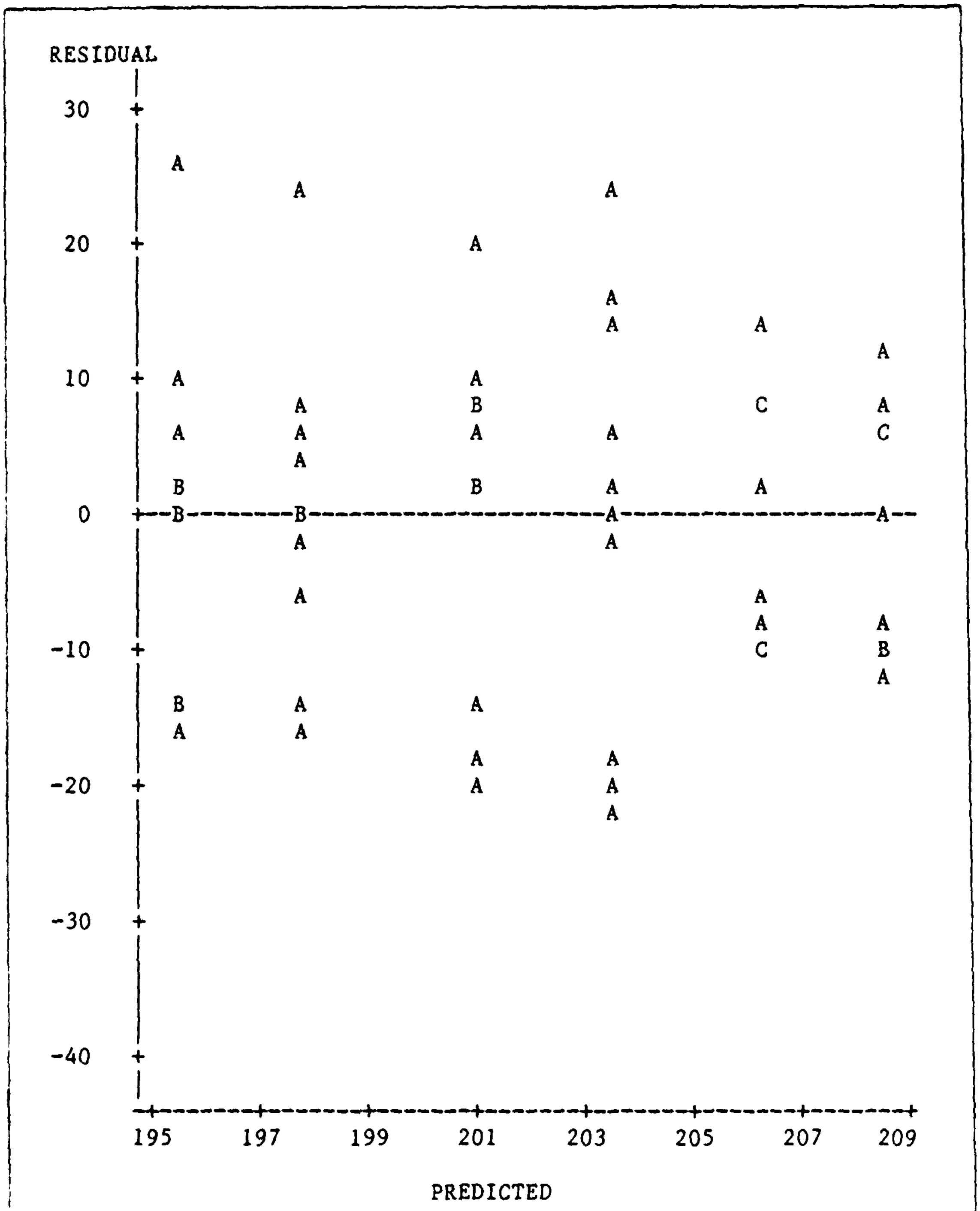


FIGURE 9. RESIDUALS FOR SUCCESSFUL INFILTRATIONS.

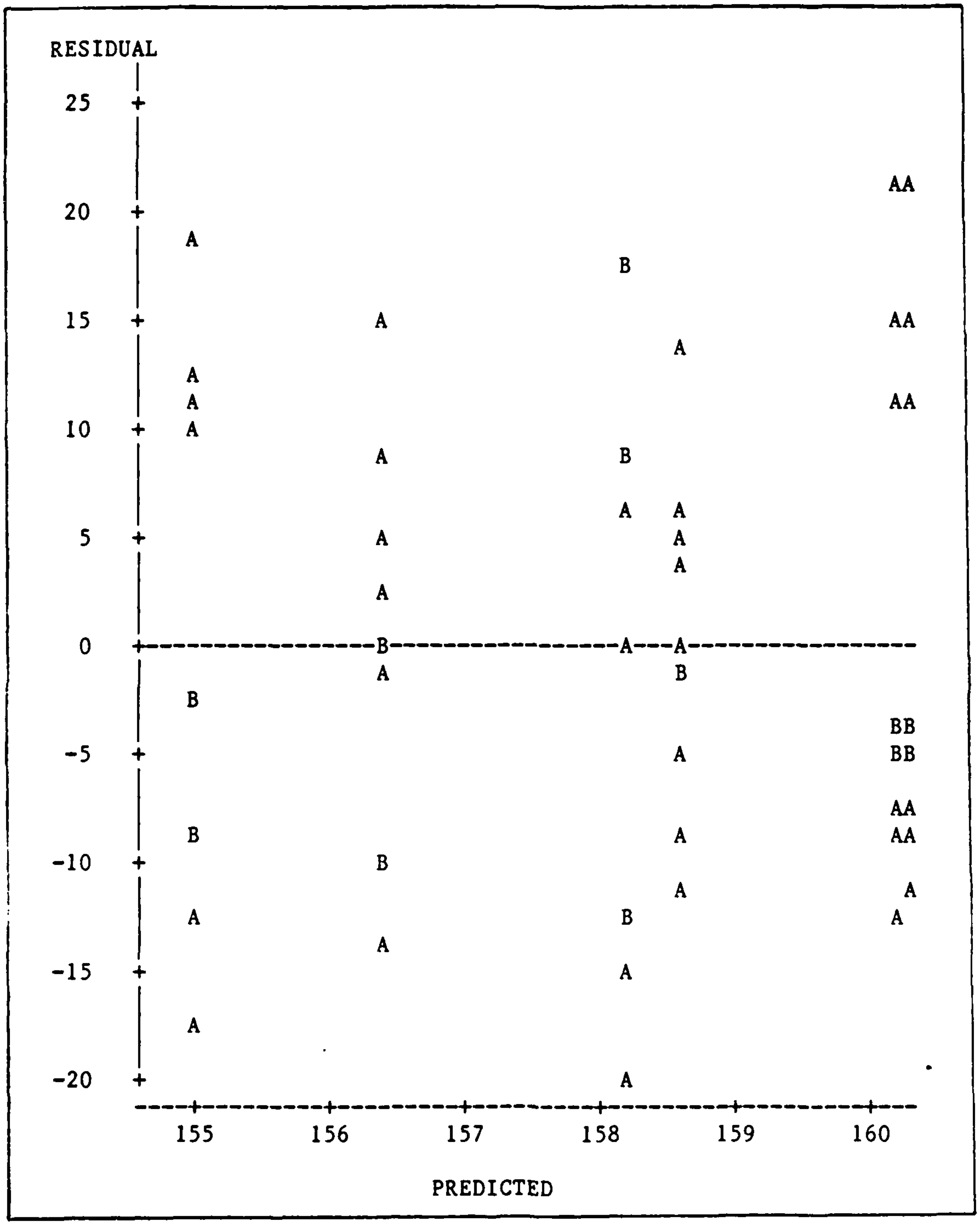


FIGURE 10. RESIDUALS FOR SUCCESSFUL EXFILTRATIONS.

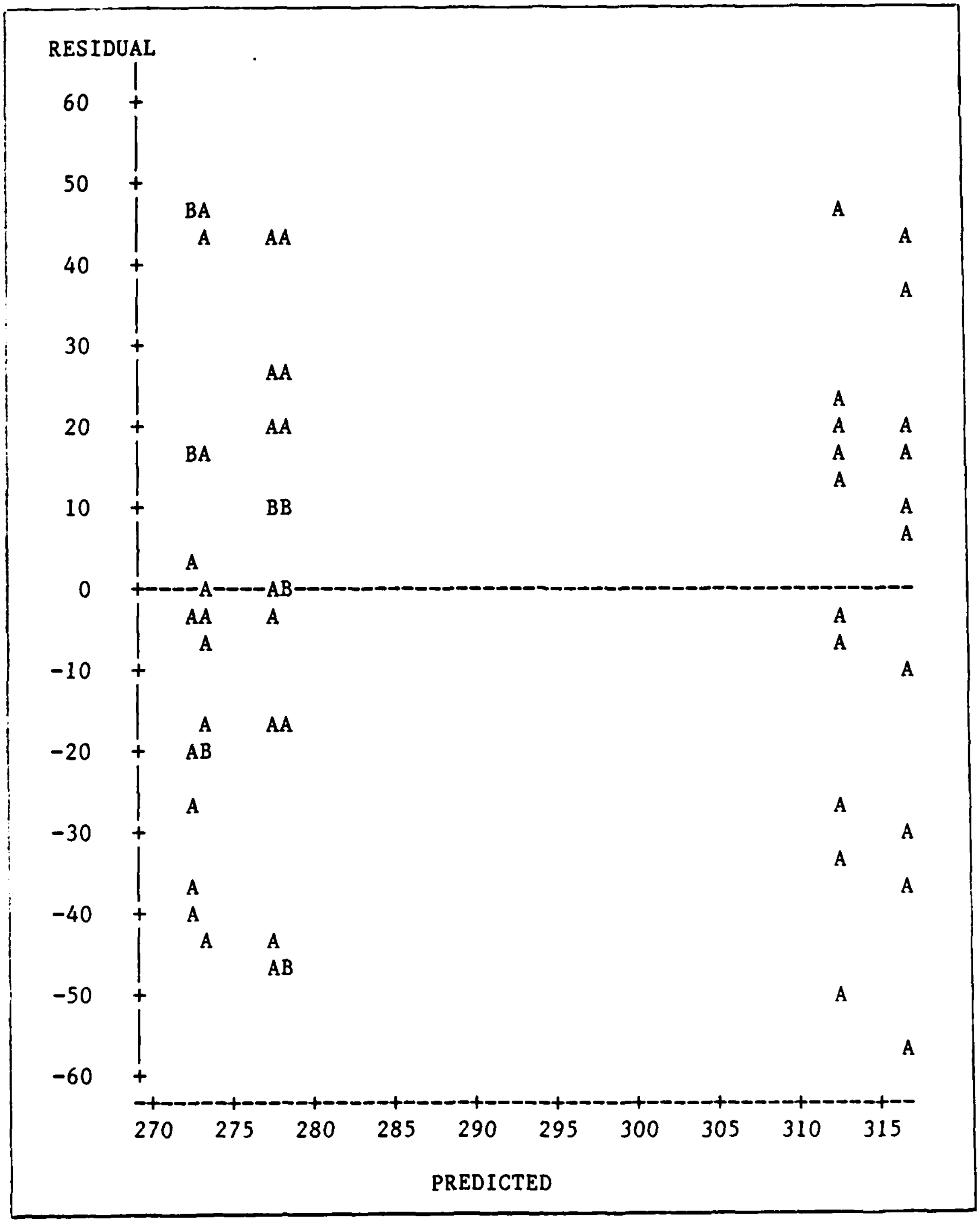


FIGURE 11. RESIDUALS FOR SUCCESSFUL RESUPPLIES.

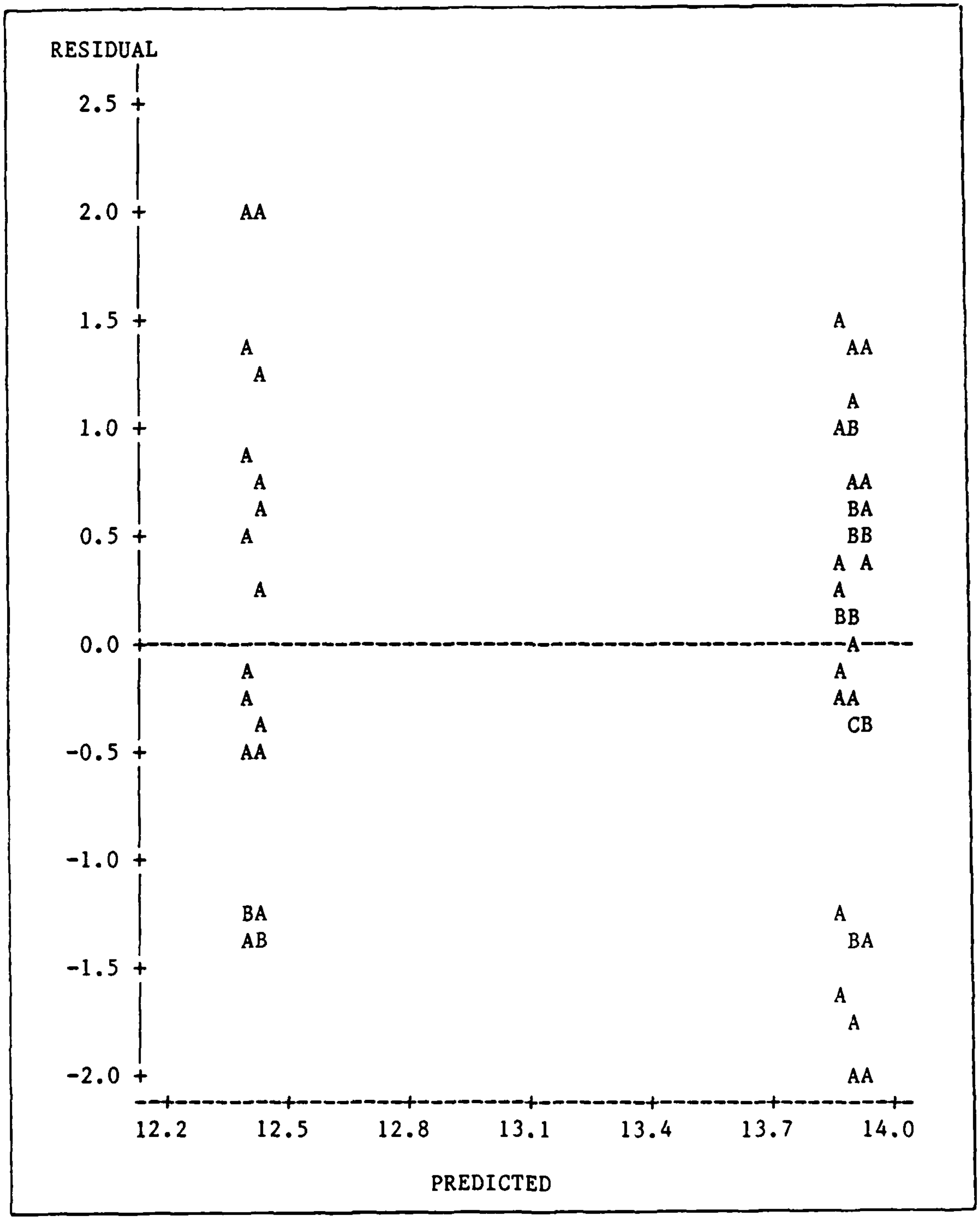


FIGURE 12. RESIDUALS FOR INFILTRATION DELAY.

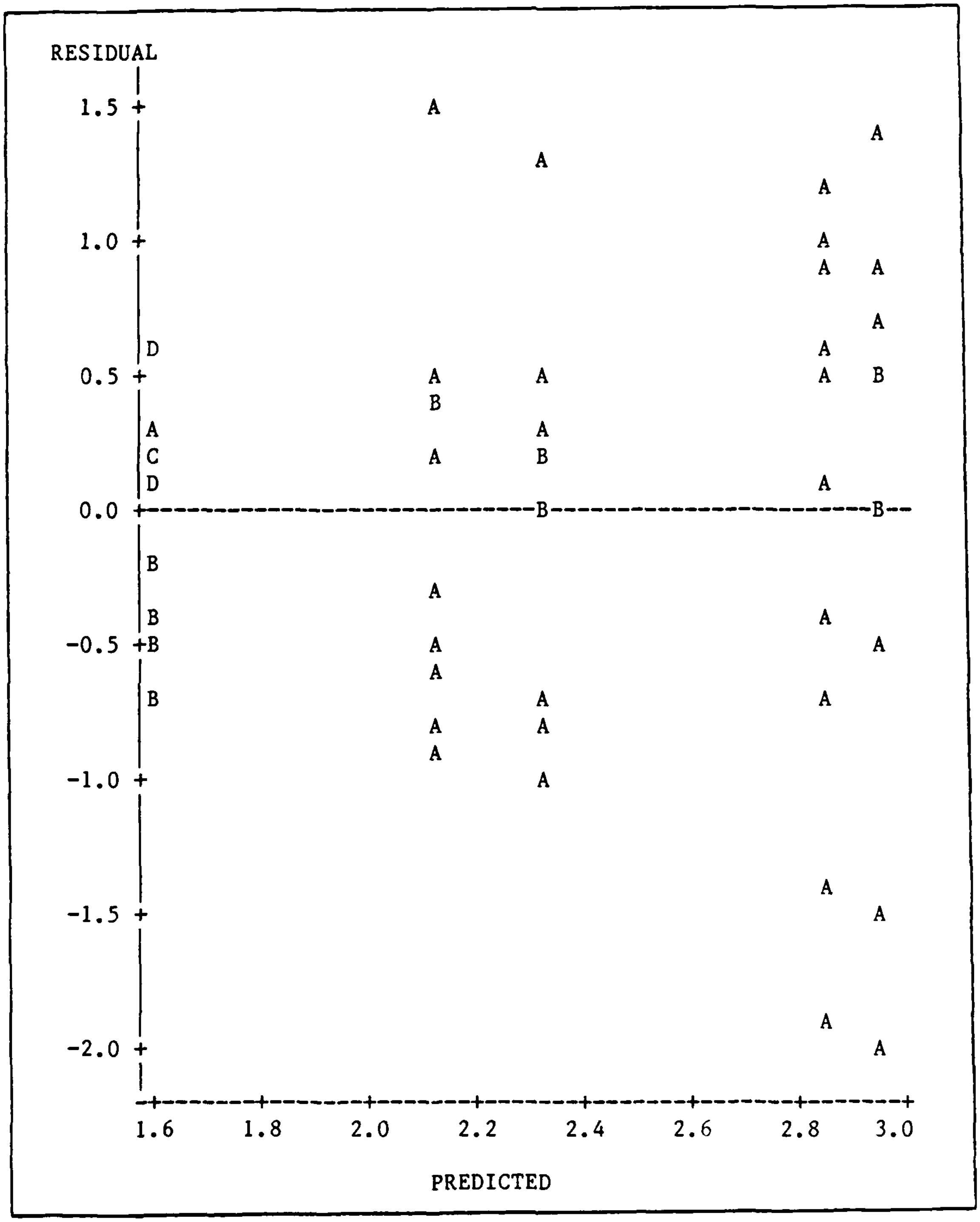


FIGURE 13. RESIDUALS FOR EXFILTRATION DELAY.

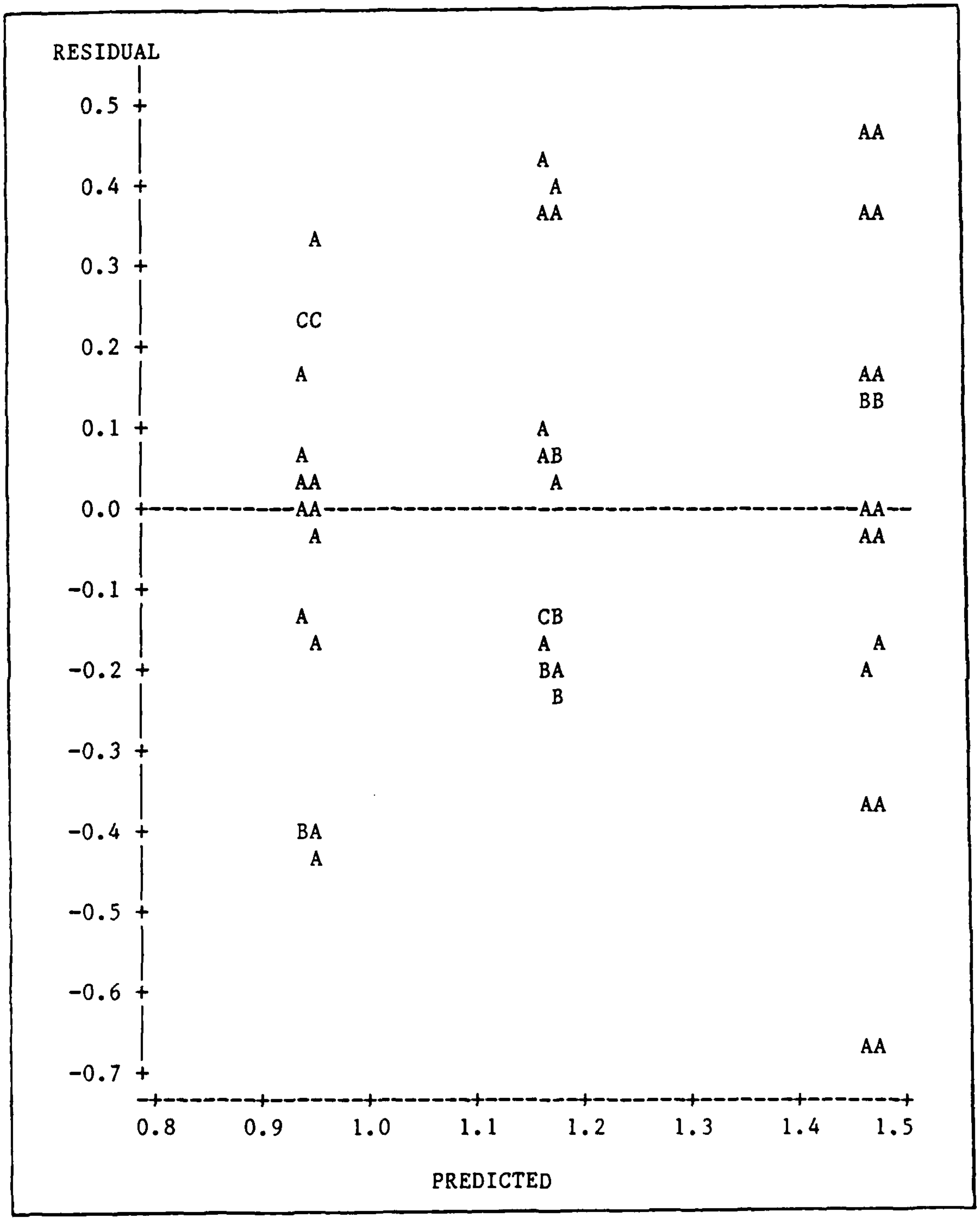


FIGURE 14. RESIDUALS FOR RESUPPLY DELAY.

Regional Priorities

In addition to the previous seven response variables, the different basing option were compared on the basis of actual regional priorities. The target location data base includes desired regional priorities (See Table VII). Mission requests are prioritized according to region priorities and mission type priorities. How well a basing strategy meets the input regional priorities is a measure of its usefulness. The actual regional priorities obtained by using a given basing strategy can be determined by comparing the number of mission requests to the number of successful missions for each mission type for each region. For each region, the percent of infiltrations completed was doubled and added to the sum of the percentages of the other mission types completed. The regions were then ranked for each run. The rankings were added across all runs to get an average ranking for the basing option. The three options were then compared to the input priorities and the deviation from the input computed. The score for each option was the number of inversions required to get from the input priorities to the realized priorities. Figure 15 gives an example of the scoring method. The results are shown in Table XXII.

Interpretation of Results

In a study with as many measures of effectiveness as this study has, the measures must be weighed against each other in order to choose a "best" basing option. As stated earlier, the main measure of effectiveness used here is the total number of successful missions accomplished. This can be broken down into three sub-measures. In this scenario, option 1 results in the most successful missions being accom-

REGION	1	2	3	4	5	6
INPUT PRIORITY	3	6	5	2	1	4
OBSERVED PRIORITY	5	6	4	2	3	1

REGION 1 SHOULD BE PREFERRED TO REGIONS 2, 3, & 4
IS PREFERRED TO REGION 2
=> 2 INVERSIONS

REGION 2 SHOULD NOT BE PREFERRED TO ANY REGIONS
IS NOT PREFERRED TO ANY REGIONS
=> 0 INVERSIONS

REGION 3 SHOULD BE PREFERRED TO REGION 2
IS PREFERRED TO REGIONS 1 & 2
=> 0 INVERSIONS

REGION 4 SHOULD BE PREFERRED TO REGIONS 1, 2, 3, & 6
IS PREFERRED TO REGION 1, 2, 3, & 5
=> 1 INVERSION

REGION 5 SHOULD BE PREFERRED TO ALL REGIONS
IS PREFERRED TO REGIONS 1, 2, & 3
=> 2 INVERSIONS

REGION 6 SHOULD BE PREFERRED TO REGIONS 2 & 3
IS PREFERRED TO ALL REGIONS
=> 0 INVERSIONS

THEREFORE, THIS OPTION HAS FIVE INVERSIONS

FIGURE 15. CALCULATING INVERSIONS

Table XXI. Realized Regional Priorities.

REGION	INPUT	OPT 1	OPT 2	OPT 3
1	3	5	5	5
2	6	6	6	6
3	5	4	3	3
4	2	2	2	1
5	1	3	4	4
6	4	1	1	2
	SCORE	5	6	5

plished. It also results in the most of each type of mission accomplished. The secondary measure of effectiveness used is the delay between mission request generation and mission accomplishment. Option 1 leads to the shortest infiltration delay but ranks second in exfiltration and resupply delays. In special operations, the primary goal is to insert A-teams: exfiltration and resupply are secondary. Since option 1 is the quickest in inserting teams and second in both extracting and resupply, it is considered the best basing option to use for mission timeliness. The third measure of effectiveness is the realized regional priorities. Option 1 is at least as effective in meeting the input regional priorities as the other two options. In this scenario, option 1 is the best of the three basing options considered from the perspective of all three measures of effectiveness.

Mission Delays

Mission delay is defined as the time between the generation of a mission request and actual accomplishment of the mission. When an infiltration mission request is generated, it is first presented to the Army for an A-team to be assigned. The lengthy delays for infiltrations are due to the lack of A-teams. Average delays for A-team assignment range from ten to fourteen days. Therefore, the delay between the mission request being presented to the Air Force and the successful completion of the infiltration is much less than the measured twelve to fourteen days. One possible explanation for the lack of available A-teams is the mission priority scheme. In the scenario used, all infiltration missions were attempted before any exfiltrations. Therefore, the teams may be unavailable for further missions because they

have not been extracted yet.

Three excursions were run on this assumption. The excursions all used basing option 1. Excursion 1 prioritized infiltrations and exfiltrations equally, excursion 2 placed exfiltrations above infiltrations, and excursion 3 included a dynamic feature. In excursion 3, infiltrations were prioritized above exfiltrations initially. If the team queue grew beyond fifteen mission requests, the priorities were reversed. If the team queue decreased below ten mission requests, the original priorities were reinstated. These three excursions were all compared with the original case using basing option 1. The ranking of the three excursions and the base case are shown in Table XXII. The results show that changing the relative mission priorities does not significantly reduce the waiting time for infiltration missions in this case, but it does affect the capability of the force. All three excursions improved force capability over the base case. This indicates that the priority

Table XXII. Ranking of Excursions.

RESPONSE	BASE	EX 1	EX 2	EX 3
TOTAL MISSION	[illegible]	[illegible]	[illegible]	[illegible]
INFILTRATION	[illegible]	[illegible]	[illegible]	[illegible]
EXFILTRATION	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

scheme used during the primary analysis might not be the best to implement.

Quick Response Use of the Model

The model used here is designed for use in long term planning. As such, it is not designed for quick response studies. However, it can be used in a quick response mode when questions are asked about options not considered in a completed study. For example, the study explained here considered three basing options. A possible question arising from the study being briefed is: What if Taiwan allowed us to used the airport at Taipei and we distributed the AFSOF aircraft as shown in Table XXIII?

Table XXIII. Quick Response Basing Option.

	CLARK		OKINAWA		NAGASAKI		KUNSAN		TAIPEI	
ACFT	NBR	CWS	NBR	CWS	NBR	CWS	NBR	CWS	NBR	CWS
MC-130	3	5	2	3	-	-	-	-	2	3
HH-53H	-	-	-	-	-	-	7	11	-	-
HC-130II	1	2	2	3	-	-	-	-	-	-
HC-130I	1	2	2	3	-	-	-	-	-	-
CV-22A	-	-	2	4	3	6	-	-	2	4

The subroutine OTPUT can be used to provide the quick response capability desired. The subroutine OTPUT is used by SLAM to allow the modeler to collect and output most statistics desired. In this study, it was used to print the desired mission statistics to a separate file, which then could be used for input to A . However, the regional priority data could not be obtained in OTPUT. This is because that data was contained in the SLAM histograms and could not be broken out. Therefore, the only statistics used for the quick response question con-

sidered mission accomplishment and delays across all regions. In order to answer the question posed above, the basing, weather, and special entry input data files were modified to include Taipei. The model was then run and the data output by OTPUT was appended to the output data from the primary model runs. The complete output data file was then input to SAS with the indicator variables OPT and WX added to it. Weather data was readily available for Taipei, or the question could not have been answered quickly. Total time needed for updating the data files was less than two hours. The model runs were completed overnight, and the SAS run took fifteen minutes. An additional half hour was needed to put the output from the SAS run into a form that could be briefed. Therefore, the elapsed time from being asked the question to being ready to brief the answer was less than one day. This is dependent on data and computer availability. Questions for which weather data is not readily available can be answered approximately in a short time if weather data can be approximated. The time required depends on the subroutine OTPUT collecting the proper statistics.

Summary

The three basing options studied were compared on the bases of total successful missions, successful infiltrations, successful exfiltrations, successful resupplies, average infiltration delay, average exfiltration delay, average resupply delay, and regional priorities. Analysis of variance was used to identify significant differences in mission accomplishment between the options. Data collected for all seven mission accomplishment response variables was judged to be consistent with the assumptions needed to allow using ANOVA. Regional prior-

ity was judged by taking the ratios between successful missions and requested missions and comparing the results to the input priorities. Of the three basing options, basing option 1 is the preferred option based on force capability.

The collected data showed that the average infiltration delay was much higher than the delays for the other mission types. A possible reason for the large average could be the mission priority scheme. Three excursions were run using different priority schemes. The results of the excursions showed no decrease in the average infiltration delay, but did show an increase in numbers of successful missions. This indicates the priority scheme does have an impact on force capability, but another factor, such as overtasking the AFSOF, is causing the large infiltration delay.

The model was also used to provide a quick response to a basing option question asked after completion of the initial study. The speed of the response is dependent on the availability of the weather data needed and the coding of the subroutine OTPUT. If weather data is readily available, the statistics are available from SLAM, and OTPUT is written correctly, an answer can be prepared in less than a day.

The chapters to this point have addressed this study. Chapter VI will contain observations based on this study and recommendations for future work that could be done in this area.

VI. Observations and Recommendations

Introduction

This study demonstrated a simulation of the AFSOF. The specific scenario used was a Far-East war between Communist China and the United States. The system is a combination of two interlocking models. One subsystem models the US Army airlift requirements to include: target locations; infiltration, exfiltration, and resupply rates; and A-team availability. The other subsystem models AFSOF aircraft locations, capabilities and availability; mission planning; and recycling aircraft and aircrews. Three threat levels, ranging from benign to threat requiring sophisticated threat avoidance equipment and tactics, are included. Weather in the form of ceiling, visibility, wind, rain, and turbulence, is also considered. This chapter provides observations regarding the uses and limitations of the model. It includes some ideas for further work concerning modeling of the AFSOF.

Observations

Uses. The model used was designed to aid long range AFSOF planning. It can be used to estimate the relative capabilities of different forces or of the same force under different basing options. Actual force capability cannot be estimated using the model because of the aggregation of targets and the simplifications in mission planning. The model can be used to address basing questions, as it was used in this study. It can also be used to address other "what if" questions about aircraft and aircrew alternatives. For example, it can be used to estimate the change in force capability resulting from a change in crew

ratios. One reason for its usefulness is that it collects many statistics during a run. Statistics collected include missions required, missions scheduled, successful missions, air aborts, crashes, scheduled and actual flying hours, and fuel requirements. The statistics are collected by mission type and by aircraft type. These statistics and others can be accessed and reported through modifications to the SLAM code, so the model itself need not be changed. Thus, additional information can be reported and used without any modifications to the model.

Limitations. Even though the model is designed to be flexible to increase its usefulness, it does have some limitations. The limitations are imposed because of simplifying assumptions. These limitations include aircraft modeling; the way threat, targets, and weather are modeled; and constraints placed on mission planning.

Aircraft modeling is a major limitation. The model deals at a macro level only. It does not model any aircraft subsystems, but treats the aircraft as a single entity. Therefore, it cannot be used to assess the impact of changes in instrumentation within aircraft unless those different instrument options are added to the model. Detailed modeling of this type was deliberately left out of the model, and adding the detail could require major revisions to the model.

Threat is modeled very simply. Aircraft have the same attrition rate regardless of the threat level they are exposed to. Aircraft are barred from use on missions with higher threat levels than their input capabilities. Therefore, the flexibility afforded by highly qualified aircrews is ignored.

Targets are also simply modeled due to the aggregation of the

target data base for classification purposes. Geographical features are not included, so a target may be located in a body of water or on a mountian peak. This problem could be overcome by making the target regions very small, but this could pinpoint targets, which would raise the classification of the target base above Secret.

Weather is included, but correlations between regions and correlation over time is not included. The weather at a base and the weather in a target region could be highly correlated in reality, but the model assumes they are independent. It also assumes that the weather at a base for one flight is independent of the weather for any other flight, regardless of the time lapse between flights.

A fourth area that was simplified for modeling ease was mission planning. The model assumes that all missions are flown directly from the home base to the target and back. This results in sensitive missions being flown over highly populated areas, which is very unrealistic. It also understates the mission lengths, since actual missions are usually not flown by the most direct route. Air refueling points are evenly spaced along the routes, which also may not be realistic. Another limitation in mission planning is that the model does not allow multiple targets for a single mission, when in reality closely spaced targets would be serviced by the same mission.

Other limitations are imposed by the maximum sizes of the input data bases. Currently, only ten aircraft types, ten target regions, ten aircraft bases, and five mission types are allowed. Additionally, if Combat [illegible] [illegible] missions and [illegible] missions are flown to the same regions, the threat distributions for the different missions must be the

same. Ten regions are allowed for both CR and SOF in the target data base, but the model includes only ten regional threat distributions. Therefore, if both CR and SOF are included in the model, the region numbers assigned to CR must differ from those assigned to SOF unless the threat distributions are identical.

Recommendations for Further Study

As stated above, the model used in this study has several limitations. Improvements to the modeling of weather and targets might improve the accuracy of the model some, but potential improvement would be small compared to the work that would be needed. Weather is difficult to model accurately, and would require large data bases for support. Improved target modeling would also require large data bases. In either case, the work required to develop and to integrate the required data bases would be enormous. On the other hand, further work could be done to reduce the limitations imposed by the threat modeling and mission planning or to add capability to look at other options. Improvements in the following areas could significantly improve the model:

a. Allow for different attrition rates based on aircraft capability and threat levels.

b. Add the capability to accomplish more than one mission per sortie.

c. Add divert bases to the model.

d. Allow a mix of tanker aircraft to support the same mission.

Short Term Planning Model. This model was designed for long term planning. As a result, it is not suitable for studies with short sus-

penses. In order to use the model, the six input data bases must first be built and checked. Then the output routine would need to be modified to report the appropriate response variables. This process would require at least six to ten weeks. The quick response capability outlined in Chapter V is useful only after the data bases and data reporting routines have been built. A deterministic model would be more useful in time sensitive studies. Such a model could use location analysis, which would be geared to find the optimal basing strategy -- not just the best of several options. Or it could be an allocation model, which would assess the capability of a force under different basing options by optimally assigning missions. I investigated the feasibility of the allocation model, and it appeared to be worthwhile. The model could be completely automated with all inputs made interactively or through previously prepared data base files. If automated, it could be used in a decision support system to provide real time inputs to decision makers regarding AFSOF basing and mission allocation.

Appendix A.

CRASOF-2

COMBAT RESCUE AND SPECIAL OPERATIONS FORCES MODEL

Programmer's Manual

December 1986

The following Programmer's Manual is an updated version of the "Combat Rescue and Special Operations Forces Model Programmer's Manual" by Maj Jack Dickinson, September 1985 (3)

INTRODUCTION

The Combat Rescue and Special Operations Forces model (CRASOF) is a discrete-event simulation model written in SLAM II (Simulation Language for Alternative Modeling, by Pritsker & Associates) and supported by FORTRAN 77 subroutines. We assume that the user of this manual is familiar with SLAM, FORTRAN, and simulation in general.

A simulation run requires the user to supply six input files. The data is broken down into a relational data base structure to reduce the total number and size of input files required on line. These files are identified by the following system:

SOFAC--.DAT	Data on aircraft performance and capability. The model accommodates up to ten aircraft types simultaneously.
SOFBS--.DAT	Aircraft base locations with initial aircraft and aircrews assigned. Holds up to ten bases.
SOFEN--.DAT	Number and type of aircraft in theater including deployment dates. File also specifies length of run and accommodates parameter changes during the course of a run.
SOFTG--.DAT	Target distribution file for SOF and CR missions. Holds up to ten regions for each.
SOFWX--.DAT	Cumulative probability distribution of ceiling, visibility, winds, rain, and turbulence for the theater.
SOFXX--.DAT	Theater specific information on mission rates, distances from threat, etc. Following is a complete definition listing of the 400 data elements in the XX array. Note: the user must set only the items preceded by an asterisk (*). Changing other items may alter model operation and produce invalid results.

*XX(1)	Priority of mission type 1.
*XX(2)	Priority of mission type 2.
-	-
-	-
-	-
*XX(10)	Priority of mission type 10.
*XX(11)	Days before first infil mission.
*XX(12)	Last day for generating new infils.
*XX(13)	Days in phase 1.
*XX(14)	Days in phase 2.
*XX(15)	Days in phase 3.
*XX(16)	Required infils/day in phase 1.
*XX(17)	Required infils/day in phase 2.
*XX(18)	Required infils/day in phase 3.

*XX(19)	Required infils/day in phase 4.
XX(20)	Infil rate updated daily by model.
XX(21-30)	Not Used.
*XX(31-40)	Probability that a team requires exfil in regions 1-10. Probabilities range from 0.0 to 1.0. Set unused regions to 0.0.
*XX(41-50)	Delay in days between infil and exfil in regions 1-10 (eg. infil day 1, delay 3.0, exfil day 4).
*XX(51-60)	Delay in days between infil and resupply in regions 1-10 (eg. infil day 1, delay 2.0, resupply day 3).
*XX(61-70)	Minimum delay in days between team's exfil and infil in regions 1-10 (eg. exfil day 4, delay 3.0, infil day 7).
*XX(71-80)	Mean days aircrew can survive/evade in regions 1-10.
*XX(81)	Maximum A/R's on one mission (at least 1.0).
*XX(82)	Maximum missions spared by one tanker (at least 1.0).
*XX(83)	Maximum number of tankers on one mission (at least 1.0).
*XX(84)	Number of active bases.
*XX(85)	Flying window (up to 23.0 hours).
*XX(86)	Launch windows (waves) per day (limits max sorties/acft/day).
*XX(87)	Total active teams assigned to regions 1-10 (at least 1.0).
*XX(88)	Total reserve teams assigned to regions 1-10.
XX(89)	Model sets to number of aircraft types.
XX(90)	Model's infil/rescue mission counter.
*XX(91)	Total infil requirement over simulated period.
*XX(92)	Total exfil requirement over simulated period.
*XX(93)	Total resupply requirement over simulated period.
XX(94)	Total successful infils over simulated period.
XX(95)	Total successful exfils over simulated period.
XX(96)	Total successful resupplies over simulated period.
XX(97)	Total successful rescues over simulated period.
XX(98)	Total successful tanker missions over simulated period.
XX(99)	Unmet total SOF demand today
XX(100)	Unmet combat rescue demand today
XX(101-110)	Daily infil mission statistics:
101	Infils required
102	Infils scheduled
103	Successful infil missions
104	Air aborts on infil missions
105	Crashes during infil missions
106	Primary acft flying hours scheduled on infil missions
107	Actual primary acft flying hours on infil missions
108	Tanker flying hours scheduled on infil missions
109	Actual tanker flying hours on infil missions
110	Not used. May be programmed for [illegible]
XX(111-120)	Daily exfil mission statistics [illegible]
XX(121-130)	Daily resupply mission statistics [illegible]
XX(131-140)	Daily rescue mission statistics [illegible]
XX(141-150)	Daily tanker mission statistics [illegible]

141	Tanker missions required.
142	Tanker missions scheduled.
143	Successful tanker missions.
144	Missions canceled due to tanker crash.
145	Tanker crashes.
146	Flying hours scheduled on tanker missions.
147	Actual flying hours on tanker missions.
148	Scheduled fuel offload in lbs.
149	Actual fuel offload in lbs.
150	Not used. (May be programmed for tankers if needed.)
XX(151-160)	Daily Type 1 aircraft statistics:
151	Aircraft available at start of flying day.
152	Sorties scheduled.
153	Successful sorties (ie. mission accomplished).
154	Scheduled flying hours.
155	Actual flying hours.
156	Scheduled A/R's.
157	Actual A/R's.
158	Scheduled fuel onload during A/R's (lbs).
159	Actual fuel onload during A/R's (lbs).
160	Crashes.
XX(161-170)	Daily Type 2 aircraft statistics (like 151-160).
-	-
-	-
-	-
XX(241-250)	Daily Type 10 aircraft statistics (like 151-160).

Model uses a combination of XX(251-267) values to rank aircraft alternatives. Warning: Only aircraft with positive total scores are feasible alternatives!

XX(251)	Bonus factor for no A/R required.
XX(252)	Max bonus factor for no A/R required (MSNCR/ACCR is also added).
XX(253)	Score if aircraft capability matches mission threat.
XX(254)	Score if aircraft capability exceeds mission threat.
XX(255)	Not used.
XX(256)	Score when A/R required and primary tanker available.
XX(257)	Score when A/R required and only alternate tanker available.
XX(258)	Score when primary aircraft on his top priority mission.
XX(259)	Score when primary aircraft on his 2nd priority mission.
-	-
	-
	-
[illegible]	Score when primary aircraft on his 10th priority mission.

[illegible] currently defined.

XX(268) Start search rank in file 2 for ALLOC-2.

*XX(269) Cancel resupply if exfil request in aircraft queue (yes=1.0, no= 0.0).

XX(270) Simulation time for end of daily flying window,

XX(271) Set to 1.0 if any aircraft in requirements mode else 0.0. If 0.0, model quits trying to schedule missions when all aircraft are busy.

XX(272) time between desired waves of sorties for today.

XX(273) Next time today to try flying more sorties. Allows aircraft on short missions or in maintenance to fly again.

XX(274) On/off flag for event 5 (on=1.0, on=0.0).

*XX(275) End surge adjustment in days for tankers added to item 18 in SOFAC--.DAT.

XX(276) ALLOC() on/off switch (on=1.0, off=0.0).

*XX(277) Data echo switch (none=0.0, XX() only=1.0, all=2.0).

*XX(278) Days before rescue flies type 1 threat.

*XX(279) Days before rescue flies type 2 threat.

*XX(280) Days before rescue flies type 3 threat.

*XX(281) Number of days before first rescue mission.

*XX(282) Last day for generating new rescue missions.

*XX(283) Days in rescue phase 1.

*XX(284) Days in rescue phase 2.

*XX(285) Days in rescue phase 3.

*XX(286) Required rescues/day in phase 1.

*XX(287) Required rescues/day in phase 2.

*XX(288) Required rescues/day in phase 3.

*XX(289) Required rescues/day in phase 4.

XX(290) Rescue rate updated daily by model.

XX(291-300) Not Used.

Note: XX(301-320) apply to both SOF and rescue. If both SOF and CR are flown in the same region, the probability distributions must be equal.

*XX(301-310) Probability of type 1 threat in regions 1-10.

*XX(311-320) Probability of type 1 or type 2 threat in regions 1-10.

XX(321-330) Not used.

XX(331-340) Daily resource slack (min of aircraft or crews available-sorties scheduled) for aircraft types 1-10.

*XX(341) Day to start accumulating tanker sortie statistics for printout of average sortie rates (no printout if=0.0).

*XX(342) day to stop accumulating tanker sortie statistics.

XX(343) Accumulated tanker sorties for days XX(341)-XX(342).

XX(344) Accumulated tanker sorties from day XX(342) to end of simulation.

*[illegible] Omit infils in team queue from total infils required (yes=1.0, no=0.0).

[illegible] Total infil, exfil, and resupply missions required

[illegible] today.

XX([illegible]) [illegible] today.

XX([illegible]) [illegible] today.

XX([illegible]) Active teams available today.

XX([illegible]) Reserve teams available today.

XX([illegible]) Crews available today for aircraft types [illegible]

XX(361-[illegible]) Aircraft in theater today for types [illegible]

*XX([illegible]) Army does all [illegible] missions [illegible] range [illegible] from [illegible]AF beddown.

*XX([illegible]) Army may do [illegible] missions [illegible] range [illegible] from [illegible]AF beddown.

*XX([illegible]) Probability Army does infil from [illegible]

*XX([illegible]) Probability Army does exfil from [illegible]

*XX([illegible]) Probability Army does resupply from [illegible] XX([illegible]).

XX(376-379) Not used.

*XX(380) Minimum crew rest in hours.

*XX(381-383) Mission 1-3 preparation time in hours.

*XX(384) Low threat preparation time for CR or tanker mission.

*XX(385) Medium to high threat preparation time for CR or tanker mission.

XX(386-390) Not used.

Note: XX(391-400) control FORTRAN calls activating SLAM RECORD statements from file SOFCD--.DAT. Enter RECORD statement number to activate SLAM table/plot (max of 10). Enter 0.0 in unused locations.

*XX(391) 1st SLAM RECORD statement number.

*XX(392) 2nd SLAM RECORD statement number.

\- \-

\- \-

\- \-

*XX(400) 10th SLAM RECORD statement number.

NETWORK DESCRIPTION

The SLAM portion of the model starts with a group of STAT and RECORD statements which identify variables to be printed in the final run report. The PRIORITY statement indicates that missions have priority in the crew and aircraft queue according to ATRIB(21) which is set to the values in XX(1-10). This permits user control over the servicing of the various mission types.

Next, the various army team, aircraft, and aircrew resources are identified. The numbers which follow each RESOURCE statement indicate the queue number which has most priority for drawing on that resource.

Following these statements are eight short blocks of code which govern the CR/SOF network. The first section starts flying activity each day and then launches a number of 'waves' throughout the day. A wave is a time at which all waiting missions are matched with the best available aircraft and corresponding crews. At the end of a flying day this segment collects and resets statistics as required.

The next block merely sets the starting and ending days of activity for SOF and rescue. Following this, at label NEW1, is the code which generates infiltration missions according to the user-specified rates in array XX. Several of the mission transaction attributes are set and then the mission proceeds to the team queue (queue 1) according to the logic in ALLOC-1, which is a FORTRAN subroutine. If the infil is an Air Force responsibility, the mission proceeds to the aircraft and aircrew queue (queue 2) according to the logic in ALLOC-2. Following this queue (labeled QUE), the mission proceeds to EVENT-7, which is a FORTRAN routine that generates resupplies and an exfil as necessary.

The combat rescue code starts at label NEW2 and the logic parallels that of the SOF infiltration code except that no army teams are required from the team queue.

The next block contains several small resource handling routines. Code is provided for flying and recycling the aircraft and crews. The block of code at label WDLY recycles resources in the event a weather cancellation occurs. The section starting at the UTE label draws aircraft as required to prevent the funded ute rates from being exceeded. The logic for this action is governed by ALLOC-3. Lastly, the code at the ALTR label withdraws resources from the theater for redeployments as specified by the user in the ENTRY file (SOFEN--.DAT).

The INIT statement sets simulation end time to a dummy value of 99. The actual end time is set by the user in the ENTRY file so that no changes need be made to the SLAM network in the course of a study. SEEDS statements provide up to ten sets of random number seeds for the ten streams used by the model.

ATTRIBUTE DEFINITIONS

Attribute definitions for the mission transaction are defined as follows:

Attribute	Definition
1	The region containing the primary mission objective.
2	The simulation time that the mission request arrives in the system.
3	The combat radius of the mission.
4	The threat type facing the primary aircraft.
5	The type mission (Infil=1, Exfil=2, Resupply=3, Rescue=4, Refueling=5).
6	The mission number. SOF missions supporting the same team have the same mission number.
7	Tanker flying hours.
8	Primary aircraft flying hours (converted to days for use in the SLAM network).
9	SOF teams assigned to the mission.
10	The primary aircraft type for this mission.
11	The number of primary aircraft on the mission plus .01 times the home base for the primary aircraft.
12	The tanker aircraft type supporting the mission.
13	The number of tanker aircraft supporting the mission.
14	Mission success flag (0.0 means successful; 1.0 means sortie is unsuccessful).
15	Team status flag (0.0 means successful; 1.0 means team died).
16	Simulation time for last tanker landing.
17	Primary aircraft that crash on this mission.
18	Tanker aircraft that crash on this mission.
19	Weather delay for this sortie.

20	Predicted simulation time for capture of evading aircrew for rescue missions.
21	Ranking attribute for aircraft and crew mission file (file 2) based upon mission time and XX inputs.
22	Total primary aircraft aircrews assigned to the mission.
23	Number of primary aircraft aircrews that crashed and are assumed dead for now.
24	Total tanker aircraft aircrews assigned to this mission.
25	Number of tanker aircraft aircrews that crashed and are assumed dead for now.
26	The type aircraft for rescued CR/SOF aircrew picked up on this mission.
27	The number of CR/SOF aircrews rescued by this mission plus .01 times the home base for the rescued crews.
28	The auxiliary attribute pointer for missions with tankers.
29	The target latitude in degrees north.
30	The target longitude in degrees east.

AUXILIARY ATTRIBUTE DEFINITIONS

Aux Attribute	Definition
1	Home base for first tanker.
2	1 plus .01 times the number of crews on first tanker.
3	Number of tankers killed plus .01 times the number of crews downed on first tanker. (Negative for creation of assets.)
4	Home base for second tanker.
.	
.	
.	Continues through maximum number of tankers (XX(83).

These attribute definitions only apply to mission transactions. For the ENTRY input file, EVENT scheduling, and special purpose files, the attribute assignments are discussed under the specific subroutine code.

PROGRAM MAIN

Program MAIN sets dimensions for key variables and logical unit numbers for the simulation language, SLAM. CRASOF-2 variable dimension differ from the standard SLAM program MAIN. CRASOF-2 dimensions arrays NSET and QSET at 30,000 and sets NNSET at 30,000. In addition, CRASOF-2 dimensions array XX at 400, which requires corresponding changes to the simulation language. Within SLAM, redimension array XX in common SCOM1 to 400; redimension array XXI in common GCOM1 to 400; set the global variable limit MMXXV to 400. CRASOF-2 does not run without these modifications to SLAM.

The logical unit numbers in program MAIN coincide with the standard SLAM values. SLAM simulation program (cards) via unit 5. SLAM writes to unit 6. SLAM requires unit 7 as a disk "scratchpad." Although not shown in program MAIN, CRASOF-2 uses units 1-4, 8 and 9 for data input and units 10-19 for recording observations during the simulation. CRASOF-2 uses unit 20 for a special output file. Future modifications must avoid use of unit numbers 1-20 or make corresponding changes within CRASOF-2.

SUBROUTINE ALLOC (IALL, IFLAG).

Subroutine ALLOC allocates the team, aircraft, and crew resources according to the encoded rules. The advantage of consolidating the allocation into FORTRAN routine is that much more complex decision algorithms are possible. ALLOC gives access to all resources via the SLAM resource number. Resource 1 is active teams. Resources 2 is reserve teams. Resources 3-12 are aircraft types 1-10. Resources 13-22 are crews for aircraft types 1-10. Resource 23 is a dummy resource used to cause SLAM to execute portions of this routine when it would not normally do so. The resources available to each subsection of ALLOC depend upon the files listed after the resource definitions on the SLAM cards. The order in which the files are listed determine the priority of the ALLOC subsection for the resources when more than one subsection allocates the same kind of resource for different purposes. There are four subsections in subroutine ALLOC.

The first section, ALLOC(1, IFLAG), assigns teams to SOF missions. The second section, ALLOC(2,IFLAG), assigns aircraft and crews to mission requests. The third section, ALLOC(3, IFLAG), seizes aircraft for maintenance when the authorized UTE rate is exceeded. The fourth section, ALLOC(4, IFLAG), permits the user to redeploy resources during the run.

Input parameters: The input parameter IALL corresponds to the section of ALLOC being executed.

Output Parameters: The output parameter IFLAG tells SLAM whether the contents of the SLAM file associated with the ALLOC section need altering. IFLAG of zero indicates SLAM should take no action. A positive value in IFLAG indicates SLAM should remove the file entry ranked IFLAG, update the attribute values to the current values in array ATRIB, and send the entry into the SLAM network. A negative value in IFLAG removes the entry ranked IFLAG and sends it into the SLAM network without updating the attribute values.

Commons: ALLOC uses COMMON statements SCOM1, UCOM0, UCOM1, UCOM3, and UCOM4. Due to the length of the ALLOC code and differences between subsections, the write-up defers specific uses of variables in the commons to the applicable subsection methodology discussions.

Methodology, ALLOC(1, IFLAG): The first section, ALLOC-1, allocates active SOF teams (ATRIB(9)) to the requesting mission if resources are available. The routine does not use reserve teams to fill in for active teams to make up a shortage. SLAM associates ALLOC-1 with file number 1.

Methodology, ALLOC(2, IFLAG): The second section, ALLOC-2, is the heart of the simulation allocating aircraft and crews to mission requests. SLAM associates ALLOC-2 with file number 2. Since SLAM has several routines that also handle resources, the first thing ALLOC-2 does is check to see if resource creations are still in progress. If a

mission needs a tanker and two crews, SLAM automatically tries to employ the aircraft upon creation of the tanker without waiting for creation of the crew. ALLOC-2 would conclude that more tanker crews are needed if allowed to continue before SLAM completes all resource creations for the current mission. For efficiency, the current mission description resided in location 1 of ALLOC-2's file until the necessary creations occur. The ALLOC code uses SLAM pointers within the files to improve efficiency when working with only a part of a mission's attributes.

Next, ALLOC-2 checks to see if the simulation time is still within the launch window in XX(270). If not, ALLOC-2 quits.

Next, ALLOC-2 checks where the search left off upon the last attempt to use the resources for the current wave and begins at the next file entry. ALLOC-2 quits after evaluating all missions once for this wave. The search begins at the top of the file for each wave.

ALLOC-2 finally begins the search for aircraft to satisfy the mission request at this point. ALLOC-2 assumes only one primary aircraft flies the mission; however, that aircraft may require multiple tankers. The aircraft search begins with the first and proceeds through the last one defined in array ACDATA. The number of aircraft available for use by ALLOC-2 accumulates in NACQ2 so ALLOC-2 can recognize when lack of any aircraft makes further mission request evaluations futile.

If the aircraft can do this type mission in the anticipated threat environment and has sufficient crews available, ALLOC-2 begins scoring this aircraft type against the current mission request as a feasible alternative.

SCORE2 accumulates the score for the aircraft type currently being evaluated. SCORE1 will have the best score observed so far for this mission. Positive scores represent feasible alternatives. Permission to create crews and aircraft is just as good as having assets; however, actually making the new resources only occurs if the highest scoring alternative requires the new assets.

The higher the mission type is on the aircraft's priority list, the higher the mission priority score (XX(258-263)). Aircraft whose threat capabilities perfectly match the anticipated threat increment the score by XX(253) while a more threat-capable aircraft than the mission requires increments the score by XX(254). Using this heuristic scoring method, ALLOC-2 has a slight preference for perfectly matching aircraft to threat; however, ALLOC-2 recognizes that more capable aircraft can still do the mission so both options are feasible. The bottom line assumption is that the missions are stacked in the file according to their importance to the theater commander; therefore, ALLOC-2 will accomplish them in order, if assets permit. The next question in the scoring algorithm is whether A/R is necessary for this aircraft to perform the mission.

If no refueling is necessary, the routine computes a bonus incre-

ment for the score based upon the ratio of the aircraft combat radius to the mission combat radius (ARRAT). The heuristic assumes that using an asset as close as possible to its combat radius is ideal employment for that asset since it its likely to reduce overall tanker requirements as other missions are evaluated. Since the combat radii of the potential aircraft are often similar, a single bonus factor times this ratio would either not discriminate between choices at short range or become the only real factor driving the decision as the mission radius approaches the aircraft combat radius. Therefore, a maximum bonus for no A/R is stored in XX(252). XX(251) is set to differentiate between aircraft at short ranges.

To prevent aircraft with different combat radii from having the exact same score increment when using the XX(252) maximum, ALLOC-2 adds the ratio ARRAT to the score. Missions falling into this category have their complete best score (SCORE1).

Aircraft that are capable of inflight refueling continue the scoring process when the mission is beyond the aircraft combat radius. ALLOC-2 uses ARSPOT to calculate the receiver A/R requirements. If the number of A/Rs needed by this aircraft is within the maximum permitted (NARLMT), ALLOC-2 has ARTANK attempt to find tankers for the mission. If ARTANK returns ITKFLG at 2 or less, tanker support is available and ALLOC-2 continues to score this aircraft type for the mission.

Tanker scores are currently used as penalty scores with a smaller penalty for using an aircraft whose primary mission is being a tanker than for using an alternative tanker. XX(256) holds the maximum penalty for using a primary tanker; XX(257) holds the maximum penalty for using an alternative tanker. ALLOC-2 multiplies the penalty by the ratio of the number of tankers required to the number of tankers permitted. This gives the ALLOC-2 the ability to differentiate between tankers with differing offload capabilities if the number of tankers required to support the mission differs between the alternatives. ALLOC-2 assumes an aircraft needing only one A/R is a better choice than an aircraft requiring more than one A/R, assuming all other capabilities are equal. For this reason, ALLOC-2 also subtracts from SCORE2 the ratio of the number of A/Rs required by the receiver to the maximum number of A/Rs permitted.

If ARTANK returns ITKFLG as 2 or more, tankers cannot support the mission. For ITKFLG equal 2, capable tankers are defined but not available and cannot be created; therefore, ALLOC-2 keeps track of the additional tankers needed (NSHORT) and the tanker type chosen (NTKSHT). ALLOC-2 uses these variables to record a potential tanker shortfall. The reason this mission is infeasible for the aircraft is lack of tanker support. ALLOC-2 records only the first mission rejected for lack of tankers during the selection process. Subsequent evaluation demonstrated that this is an unreliable way to forecast the tanker requirement. Assume, for example, that a helicopter is available and is the best choice for an exfil that is the top priority mission. Since no tankers are available, ALLOC-2 sends an MC-130 on a lower ranked infil

mission. The model recognizes that assets remain and more requests are in the queue so it calls ALLOC-2 again. The routine picks up the search with the next mission after the MC-130 infil. Assume the same helicopter is the best choice but again lacks tanker support. ALLOC-2 would overestimate the requirement saying two tankers are needed to support the same helicopter. For this reason, tanker requirements are best inferred by observing any decrease in mission capabilities with a decrease in the number of tankers assigned to the theater. Using this iterative approach, the user should make a run allowing the model all the tankers it wants to create so that the system mission capabilities with no tanker constraints are available for comparison.

ARTANK returning an ITKFLG of 3 means the mission is beyond the capabilities of all defined tankers. Requiring more A/Rs than the limit permits and requiring range extension for non-A/R capable aircraft also make a mission impossible for the aircraft type under consideration. Negating the SCORE2 for these cases insures ALLOC-2 will always recognize them as infeasible.

ALLOC-2 retains only the highest scoring alternative. If the aircraft type just evaluated is the best so far, ALLOC-2 assigns the key information to the mission attributes or to variables ending with a "1" since this solution "won" the comparison. ALLOC-2 then looks at the next aircraft which does this type mission and repeats the scoring process.

After ALLOC-2 examines all capable aircraft, a positive SCORE1 indicates that the mission being examined gets scheduled to fly. If no new resources are necessary, ALLOC-2 begins the prelaunch sequence.

Before launching, ALLOC-2 sets IFLAG to the positive rank of the mission within the mission file so SLAM will remove it and update the attributes upon departure. ALLOC-2 places tanker base and crew attributes in the auxiliary attribute buffer, AUXAT, if the mission requires tanker support. ALLOC-2 then seizes the aircraft and crews assigning them to this mission, which prevents their use by anyone else. ALLOC-2 uses STATS(1) to place observations on the scheduled sortie in the array XX for statistics desired by the user.

ALLOC-2 also collects a statistic on the type aircraft selected for the primary mission via SLAM STAT number 42. If tankers support the mission, ALLOC-2 collects the tanker type via SLAM STAT number 46 for Threat Type I capable tankers and SLAM STAT number 47 from Threat Type II tankers. Since a Type I tanker can fly a Type II threat mission, ALLOC-2 also collects the anticipated tanker threat in SLAM STAT number 44. Having finished recording the schedule, ALLOC-2 determines what actually happens to this sortie.

ALLOC-2 places the weather delay, if any, from Function USERF(5) in ATRIB(19). If the delay prevents the aircraft from completing the sortie within the permitted flying window, ALLOC-2 cancels the sortie recording the aircraft type, weather canceling for CR in SLAM STAT

number 8 and for SOF in SLAM STAT number 27. ALLOC-2 sets ATRIB(16) negative so the SLAM network recognizes the weather cancel. To reduce variance, the model always pulls the same number of random numbers from the appropriate stream for every scheduled sortie. This prevents a weather cancel on this missions from affecting what happens to the next mission. Scheduled sorties use stream 2 for SOF missions and stream 5 for combat rescue. If the aircraft can still complete the mission within the flying window, a weather delay does not cancel the sortie.

ALLOC-2 has function USERF(6) check for weather or mechanical aborts on launched sorties placing a 1.0 in ATRIB(14) for aborting sorties. USERF(6) records the appropriate statistics.

ALLOC-2 then checks for mission effectiveness assuming the aircraft makes it to the objective. A 1.0 in ATRIB(14) indicates an unsuccessful mission.

Finally, ALLOC-2 calls LOSSES to determine if any aircraft crash. LOSSES takes all required actions should a crash occur. At this point, ALLOC-2 knows exactly what happens to this sortie and records the actual observations in array XX using STATS(2).

ALLOC-2 converts flying hours to days for use within the SLAM network. ALLOC-2 assumes the tankers remain assigned to this mission until the last tanker lands. The last tanker is either the longest tanker mission launched or the tanker performing the last A/R. ALLOC-2 puts the delay time for releasing the tankers in ATRIB(16).

ALLOC-2 collects the days delay between the mission request and the scheduled launch (SLAM STAT number 1 for rescue. SLAM STAT numbers 13-15 for SOF infil, exfil, and resupply.). ALLOC-2 also collects an overall probability of success for CR (SLAM STAT number 10) and for SOF (SLAM STAT number 29). At this point, ALLOC-2 is finished with this mission and drops to a normal stop after aircraft allocation to do some housekeeping on itself.

If ALLOC-2 has to make more crews or aircraft to fly this sortie, the routine makes them before returning to do all the code just discussed for launching sorties. When making new resources, ALLOC-2 uses ATRIB(14) as a counter for how many creations must occur before this sortie can go. SLAM immediately calls ALLOC-2 when each resource is created. ALLOC-2 uses ATRIB(15) to save the rank within the file of the selected mission. ALLOC-2 places the amount of new resources needed in the variables of UCOM4 for use by EVENT 11 which actually makes them. Tanker base information is placed in AUXAT, which is available to EVENT 11 via UCOM0. ALLOC-2 cannot tell SLAM to increase the resources because SLAM would make a recursive call to ALLOC-2 trying to put the resource to work. The compiler does not recognize recursive calls nested within SLAM routines, so strange errors during execution would occur. By actually leaving ALLOC-2 and doing the creations in EVENT 11, the model avoids the recursive call problem. Since SLAM automatically loads the top file entry into array ATRIB upon entering subroutine

ALLOC, ALLOC-2 saves the current decision data temporarily as the first file entry while creations are in progress. When all the creations are finished, ALLOC-2 jumps straight to the launch code described above without having to go through the scoring process.

If ALLOC-2 is unable to support the top priority mission but has remaining aircraft resources, ALLOC-2 evaluates the next highest priority mission attempting to use the available assets. Permission to make aircraft is as good as having them for evaluation purposes.

If no feasible aircraft/mission match exists, ALLOC-2 sets the start search location beyond the end of the file to avoid more searches through the file for this flying wave. The search begins again at the top of the file at the start of the next wave of sorties.

ALLOC-2 uses XX(274) as a flag to communicate with EVENT 5 which controls the ALLOC-2 calls. A 0.0 indicates ALLOC-2 was unable to launch a mission. ALLOC-2 can recognize remaining missions are impossible while still in the middle of the file if it runs out of aircraft. So knowing the rank of the last examined mission does not replace the need for the flag. A 1.0 in XX(274) tells EVENT 5 that ALLOC-2 launched a sortie so more launches may be possible during this wave.

To make sure that ALLOC-2 does not initiate another sortie until finishing actions on the current one, ALLOC-2 negates the time for the end of today's flying window (XX(270)). No sortie could complete a flight before a negative deadline, so ALLOC-2 is shut down until EVENT 5 restores the positive value.

The last code in ALLOC-2 is error diagnostics for details about any mission that tries to seize nonexistent resources. If this occurs, alterations to the model are probably permitting interruption of the creation process. When interrupted, ALLOC-2 may become confused and prematurely alter the file, eventually causing an excessive resource request on subsequent sorties, duplicate mission requests in the system, the current decision data being interpreted as a mission request, or other strange error.

Methodology, ALLOC(3,IFLAG): The third section, ALLOC-3, seizes aircraft resources for maintenance actions. ALLOC-3 has precedence over ALLOC-2 for assets except that ALLOC-3 may not interrupt the creation process of ALLOC-2. The top entry in file 2 has a negative ranking attribute (ATRIB(21)) if a creation is in progress. ALLOC-3 processes aircraft requests stored in file 3. For each file entry, ATRIB(10) contains the aircraft type, ATRIB(11) the number of aircraft to seize for maintenance, and ATRIB(7) the base at which to start looking for the aircraft. ALLOC-3 starts with the first aircraft maintenance request in file 3 and proceeds until finding a request for which aircraft are available. The model executes ALLOC-3 code before starting the flying day so that "broken" aircraft do not fly missions.

Methodology, ALLOC-4,IFLAG: Section four, ALLOC-4, reduces the resources available in theater as requested by the user in the ENTRY file to simulate redeployment of assets. ALLOC-4 takes precedence over ALLOC-2 unless a creation is in progress (see discussion in ALLOC-2 above, if necessary). In fact, requests stored in ALLOC-4's file 4 have the highest priority for available resources, as shown by the sequence of file numbers on the SLAM RESOURCE definition cards in file NET. ALLOC-4 decides what type resource to reduce by the associated attribute number of the request. Aircraft reduction requests have the type aircraft in ATRIB(3) and desired reduction in ATRIB(4). Crew reductions have the type aircraft flown by the crew in ATRIB(7) and desired reduction in ATRIB(8). In both cases, the base involved is recorded in ATRIB(5). If any resources are available at the designated base, ALLOC-4 releases the request without change by returning the negated rank in IFLAG tell SLAM to use EVENT 12 and reduce the resource.

FUNCTION ARCCOS(X)

Function ARCCOS computes the arccosine of a real number between -1.0 and 1.0, inclusive for use by the Function DISTANCE.

Input Parameters: The input parameter X must be a real number between -1.0 and 1.0, inclusive. If it falls outside these bounds, ARCCOS outputs an error message and returns a value of 0.0.

Output Parameters: None. As a function, ARCCOS is its own output parameter. ARCCOS takes on a value between 0.0 and pi.

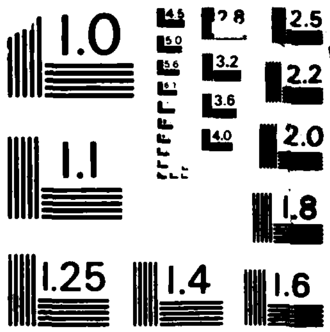
Commons: None.

Methodology: ARCCOS uses the formula

$$ARCCOS(X) = (PI/2) - ARCTAN(X)/(1-X^2).$$

AD-A179 422 BASING THE US AIR FORCE SPECIAL OPERATIONS FORCES(U) 2/2
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING M E KRAUS DEC 86 AFIT/GOR/OS/86D-6

UNCLASSIFIED F/G 15/5 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

SUBROUTINE ARSPOT (NACTYP,NAR,NARMAX,CRMSN,NHOME)

Subroutine ARSPOT computes the A/R schedule based upon the needs of the receiver aircraft. The routine builds the first three columns of array AR with each row corresponding to an inflight refueling. Column on of AR is the distance flown by the receiver aircraft since his last topoff. ARSPOT assumes the receiver starts the flight full. The second column is the onload in pounds required by the receiver to replace the fuel burned on this leg. The third column is the maximum distance from home required of the tanker to perform the refueling. Columns ten and eleven are the latitude and longitude of the refueling point. Later, subroutine ARTANK will try to match tankers to the receiver requirements generated by ARSPOT.

Input parameters: The input parameter NACTYP is the receiver's aircraft type which ARSPOT uses to obtain the performance data from array ACDATA. CRMSN is the combat radius of the mission given the current primary aircraft and its home base. NHOME is is the home base of the primary aircraft currently under consideration by ALLOC-2.

Output Parameters: The output parameters are NAR and NARMAX. NAR returns the number of A/Rs required by the receiver on this mission. NARMAX identifies the particular A/R that is most demanding for the tanker. The routine assumes the most demanding leg for the tanker is the one that makes the tanker fly the farthest from home station.

Commons: ARSPOT uses COMMON statements SCOM1, UCOM0, UCOM1, and UCOM3. SCOM1 supplies the mission characteristics in array ATRIB. UCOM1 supplies the aircraft performance data in array ACDATA. UCOM3 receives the proposed refueling schedule in array AR.

Methodology: ARSPOT begins by assigning mnemonic variable names. FLOWN is the number of miles traveled by the receiver. NAR is the number of A/Rs. ACCR is the unrefueled combat radius of the receiver aircraft. TWOACR is twice the unrefueled radius of the receiver aircraft. ARDIST is the preferred distance for the receiver to fly between refuelings based upon the ratio specified by the user in array ACDATA. ACFF is the receiver aircraft fuel flow in pounds per hour. ACTAS is the receiver true airspeed during the route. CRMSN is the combat radius of the mission. CRDIFF is the difference between the mission combat radius and the unrefueled combat radius of the receiver. ONLOAD is the pounds of fuel the receiver uses to fly ARDIST. HLAT and HLON are the coordinates of the primary aircraft home base. TLAT and TLON are the coordinates of the target. Having set the variables, ARSPOT begins to calculate the receivers requirements.

ARSPOT tries to minimize the number of refuelings while insuring the receiver refuels as far as possible from the objective area. If the mission combat radius is within 1.25 times the receiver combat radius, ARSPOT plans only one refueling on the way home.

ARSPOT plans to refuel the receiver after ARDIST unless the refueling would occur closer than the receiver's combat radius from the objective. Many of the mission aircraft must hover at the objective, which makes them want to refuel far from the objective to reduce the aircraft weight during hover. In general, aircraft refuel at altitudes higher than the normal en route altitude making them more susceptible to radar detection. For these reasons, the model insists upon the final A/R before the objective falling at the receiver's combat radius from the objective even if this results in a very small onload. If the required number of A/Rs ever exceeds the limit in NARLMT, ARSPOT assumes the mission is beyond the receiver's capabilities and quits trying to build a schedule. The maximum distance from home for the tanker while the receiver is en route to the objective is the end of the refueling track. Array ACDATA contains the required refueling track for each receiver.

After the objective, the maximum distance from home for the tanker is the beginning of the refueling track where the rendezvous begins. ARSPOT performs the first refueling on the way back home as far from the objective as possible. Since all aircraft refuel at their combat radius before objective, the tanker rendezvous occurs at the receiver combat radius minus the required A/R track from the objective. Since this is the longest distance from home for the tanker and the largest onload, this A/R is the most stringent tanker requirement or NARMAX. Once the receiver is within twice its combat radius of home station, ARSPOT assumes the receiver needs no more refueling to get home and quits.

At each refueling point, ARSPOT calculates the coordinates of the rendezvous and places them in columns 10 and 11 of the array AR.

SUBROUTINE ARTANK(NAR,NARMAX,NACTYP,NTKTYP,NTKAVL,NTKRES,MPRITK,NEWTK, NTNKRQ,ITKFLG,TKHRS,SHORT,NTCREQ,NEWTCR).

ARTANK searches for a tanker that can meet the receiver's refueling requirements contained in array AR columns 1-3, 10, and 11 as defined in subroutine ARSPOT.

Input Parameters: Input parameters include NAR, NARMAX, and NACTYP. NAR is the number of A/Rs required by the receiver aircraft. NARMAX is the most demanding A/R for the tanker. NACTYP is the receiver's type of aircraft in array ACDATA.

Output Parameters: The output parameters include NTKTYP, NTKAVL, NTKRES, MPRITK, NEWTK, NTNKRQ, ITKFLG, TKHRS, SHORT, NTCREQ, and NEWTCR. ARTANK returns the selected tanker aircraft type in NTKTYP. ARTANK puts the number of these tankers available to fly in NTKAVL. The routine places the SLAM resource number for the tanker in NTKRES. The priority of a tanker mission among the feasible missions for the selected tanker goes in MPRITK. Since the user can give the model permission to make new aircraft, ARTANK returns the number of additional new tankers the model would have to build in NEWTK. The total number of tankers needed to support the mission goes in NTNKRQ. ARTANK uses ITKFLG to describe the status of the selected tanker.

If ITKFLG is 3, the mission if beyond the capabilities of all defined tankers regardless of their availability. If ITKFLG is 2, at least one tanker is capable of the mission; however, none are available and the user did not give the model permission to make more of these capable tankers.

The grand total of the tanker flying hours required goes in TKHRS. The total number of pounds of fuel offloaded to the receiver goes in SHORT. The number of tanker crews required for the selected tanker goes into NTCREQ. Since the user can give the model permission to make new crews, ARTANK returns the number of additional new tanker crews in NEWTCR.

Commons: ARTANK contains COMMON statements SCOM1, UCOM0, UCOM1, and UCOM3. SCOM1 provides the mission characteristics in array ATRIB. UCOM1 provides the aircraft specifications in array ACDATA. UCOM3 returns the completed A/R schedule in array AR. ARTANK adds columns 4-9 of array AR. Column 4 is the tanker number assigned to this mission that is responsible for that row's A/R. Column 4 begins as a 1 until the first tanker assigned to the mission passes all the fuel it can spare; column 4 becomes a 2 until the second tanker is finished, etc. Column 5 of AR is the total pounds of fuel burned by the tanker flying from his home station to the row's A/R, which ARTANK accumulates in OUTJP. Column 6 is the total amount of fuel offloaded or dumped by the tanker including this rows' A/R as accumulated in OFFJP. Column 7 is the fuel burned by the tanker to fly home from the maximum distance out required by this row's A/R as calculated in HOMEJP. Each row in column 8 corresponds to a tanker rather than an A/R as in columns 1-7. Row 1

in column 8 had the total flying hours for the first tanker assigned to the mission. If two tankers are needed, row 2 in column 8 has the total flying hours of the second tanker, etc. ARTANK uses column 9 of row N to record the home base and the number of crews on the Nth tanker.

Methodology: The methodology in ARTANK reflects several assumptions. ARTANK assumes it should use existing tankers before making new ones; therefore, ARTANK must remember the best solution encountered requiring new tankers until it assures no existing tankers can do the mission. If more than one tanker type can do the mission by making new aircraft, ARTANK assumes the best choice is the aircraft needing the fewest new tankers. ARTANK assumes that only one kind of tanker will support each mission. ARTANK quits searching if it finds enough available tankers with crews to do the mission.

ARTANK assumes the threat encountered by a tanker is the same as the threat for the primary aircraft.

ARTANK searches from the first priority (NBRFST) to the last priority (NBRLST) mission. Currently, aircraft may be capable of five different missions. Since ARTANK does not initially know how many tankers are needed, ARTANK calls PLANE looking for only one tanker capable of the mission whose primary mission is refueling. If PLANE says a tanker is available that can handle the mission, ARTANK initializes mnemonic variables for the tanker characteristics.

Tanker true airspeed goes in TKTAS. Tanker fuel flow in pounds per hour goes in TKFF. The unrefueled combat radius of the tanker goes in TKCR. Note: this radius would burn all the available fuel so the tanker would have none to offload, thus it decreases as A/Rs are assigned. The tanker fuel flow per nautical mile goes in TKFFNM. The total usable fuel goes in TKFUEL: the required reserve fuel for the tanker is not available for use enroute so TKFUEL does not include this reserve. HLAT and HLON are the coordinates of the tanker's home base. DISAR is set to the distance between the tanker's home base and the first refueling point. FUELRQ is the fuel required by the tanker to reach its first rendezvous, offload the required fuel, and return home. Having initialized the local variables, ARTANK checks to see if the tanker returned by PLANE can meet the fuel requirement for its first refueling; if so, ARTANK assumes the tanker can do the other A/R', if sufficient aircraft and crews are available.

For each tanker's first refueling, ARTANK makes sure that the tanker has burned enough fuel to be capable of doing the A/R. Particularly when refueling helicopters, low airspeeds at extremely heavy gross weights bring the tanker close to stall conditions; therefore, the routine simulates dumping the excess fuel from the tanker before the first A/R (DUMP).

ARTANK assumes the same tanker stays airborne and refuels the receiver aircraft until doing the next A/R would not leave the tanker enough fuel to return home. After each A/R, ARTANK checks to see if

aircrew augmentation is required. If it is, ARTANK check to see if an additional crew is available at the tanker's home base. If another crew is available, ARTANK assigns it to the mission, updating column 9 of AR for that tanker. ARTANK completes the AR entries for the first tanker as defined in the COMMON section above. If more refueling remains, ARTANK checks to make sure using another tanker would not exceed the single mission tanker limit, calls PLANE for another tanker, adds the returned tanker to this mission if one is available, and starts again with the code for a tanker doing its first A/R. ARTANK continues this process until columns 1-7 are complete in array AR for each required A/R and columns 8 and 9 for each tanker. Finally, ARTANK must evaluate the merit of using this tanker type to do the mission.

If no new tankers are required, ARTANK assumes this is the best choice and stops searching setting ITKFLG and NEWTK to zero. If supporting the mission requires new tankers and the user permits making them, ARTANK saves this solution as a potential winner in similar mnemonic variables ending in a one and in array ARFLG so the search can continue for capable existing tankers. ITKFLG is 1 to show the mission is possible with the creation of more tanker aircraft. If the only capable tanker found so far needs new aircraft and permission to make the additional aircraft is denied, ITKFLG becomes 2 and ARTANK saves the key variables in mnemonics ending with 2 for statistics if no tankers can do the mission. During the search, an option requiring fewer new tankers replaces old alternatives which also required new tankers.

Once ARTANK finds a tanker to do the mission or completes examining all tanker capable aircraft, it updates the output parameters to the best alternative. If the make new tanker alternative wins, ARTANK updates the array AR to the values saved in array ARFLG. Returning ITKFLG as 3 and setting the output parameters to 0.0 indicates no tankers can handle the mission.

SUBROUTINE COMBLK(IFN,INAR)

Subroutine COMBLK provides write statements to output the contents of the COMMON blocks for diagnostic purposes as desired by the user. Since the internal verification process is over, all of this routine is commented out in the current model. The user can activate the routine by overtyping the "C" in the beginning of the line with a blank. If reactivated, the user should review the write statements to insure they still reflect the current contents of the desired commons.

Input Parameters: The input parameters are IFN and INAR. IFN tells COMBLK which COMMON statement to write. If COMBLK is writing the refueling information in UCOM3, INAR tells COMBLK how many rows of the refueling schedule arrays AR and AR1 need to be printed.

Output Parameters: None. Output goes to the disk output file.

Commons: COMBLK includes SCOM1, UCOM0, UCOM1, UCOM2, UCOM3, and UCOM4.

Methodology: For IFN of 1, COMBLK prints the ATRIB array. For IFN of 2, COMBLK prints the auxiliary attribute array, AUXAT. Because of the size of the arrays in UCOM3, COMBLK responds to IFN of 3 by printing the candidate refueling schedule array AR and to IFN of 4 by printing the actual refueling schedule array AR1. COMBLK prints the rest of UCOM3 when passed IFN of 5. UCOM4 is printed when COMBLK is called with IFN set to 6. The data in UCOM1 and UCOM2 can be obtained by echoing the data input files at the beginning of the run, so COMBLK does not print their contents.

FUNCTION DELAY(IXX)

Function DELAY computes the delay needed for mission requests so that they arrive in the mission queue just prior to any new missions being created on the day of arrival. This gives older missions priority over newer missions with the same ranking attribute (ATRIB(21)).

Input Parameter (IXX): The input parameter IXX is the element in array XX which contains the number of days before the mission request is needed.

Output Parameters: None. For a function, DELAY serves as its own output parameter carrying the appropriate delay in days for use by SLAM.

Commons: COMMON statement SCOM1 provides the current simulation time in TNOW and the desired scheduling delay in array XX.

Methodology: DELAY makes sure the mission request arrives in the mission queue at the beginning of the flying window on the desired day to enhance the odds that sufficient resources and time remain to accomplish the mission on the desired day. DELAY insures only a positive delay in days gets passed to SLAM to avoid a fatal execution error in SLAM from trying to back up time.

FUNCTION DISTANCE(HLAT,HLON,TLAT,TLON)

Function DISTANCE computes the great circle distance between two points, H and T, on the earth's surface.

Input Parameters: HLAT and HLON are the coordinates in degrees north and east of point H. TLAT and TLON are the coordinates of point T.

Output Parameters: None. As a function, DISTANCE is its own output parameter. DISTANCE returns a distance in nautical miles.

Methodology: DISTANCE uses the law of cosines to compute the great circle distance between two geographical points. It uses the assumption that the earth is a sphere to make its calculations. DISTANCE requires the presence of the function ARCCOS.

SUBROUTINE EVENT(IFN)

Subroutine EVENT has 15 separate routines for manipulating resources and SLAM files for a broad variety of purposed. EVENT-1 frees primary mission aircraft. EVENT-2 frees tanker aircraft. EVENT-3 frees teams. EVENT-4 makes changes to the XX or ACDATA array as instructed by the user with the ENTRY file. EVENT-5 causes ALLOC-2 to try to launch missions. EVENT-6 puts mission requests into the SLAM network. EVENT-7 sets up any appropriate follow-on missions considering the outcome of the current sortie. EVENT-8 prevents the model from launching both a resupply and exfil mission for the same team on the same day. EVENT-9 sets up the system to start the flying day by counting existing mission requests, recording available resources, and designating aircraft for maintenance today. EVENT-10 summarizes the results of today's flying activities by recording tankers scheduled, checking for excessive UTE rates, recording resource limitations, recording total mission accomplishments for the day, resetting the daily statistics section of array XX, and recording capture of evading aircrews as required. EVENT-11 creates planes or aircrews as requested by ALLOC-2. EVENT-12 adjusts the resources available in theater as instructed by the user with the ENTRY file. EVENT-13 frees primary aircrews. EVENT-14 frees tanker aircrews. EVENT-15 returns rescued crews to flying duty.

Input Parameters: The input parameter IFN corresponds to the desired subsection of the EVENT code.

Output Parameters: None. The EVENT code either changes variables in COMMON statements or changes file entries to pass output to the model.

Commons: EVENT includes COMMON statements SCOM1, GCOM1, GCOM8, UCOM0, UCOM1, UCOM3, and UCOM4. Specific applications are within the subsection discussions.

Methodology, EVENT (1): The first section, EVENT-1, frees a primary mission aircraft for use on subsequent sorties. If any of the primary aircraft crashed during the mission (ATRIB (17) greater than 0.0), EVENT-1 makes sure that ALLOC-3 is turned off so the aircraft can be freed and destroyed without SLAM interrupting to put it to use. EVENT-1 leaves ALLOC-3 on when freeing surviving aircraft. EVENT-1 updates the aircraft inventory for the aircraft's home base as designated in ATRIB(11).

Methodology, EVENT (2): The second section, EVENT-2, frees tanker aircraft for use on subsequent sorties like EVENT-1 frees the primary. Basing information is available through the auxiliary attribute array, AUXAT.

Methodology, EVENT (3): The third section, EVENT-3, frees teams for subsequent use. EVENT-3 permits rescue missions for SOF sorties to return teams also. If teams die, EVENT-3 assumes the theater commander will replace the active team loss with one of the reserve teams, if

available. As with EVENT-1 and EVENT-2, EVENT-3 turns off ALLOC-3 before freeing dead teams.

Methodology, EVENT (4): The fourth section, EVENT-4, changes arrays XX and ACDATA as requested by the user with the ENTRY input file. The number fields used do not conflict with the critical fields identifying mission requests for EVENT-7 and EVENT-8. Each change has three elements. If the first element is greater than 0, the change pertains to array ACDATA. The first element identifies the type aircraft; the second, the aircraft characteristic being changed; the third, the new ACDATA value. If the first element is 0.0, the change pertains to the array XX with the second element containing the index and the third the new XX value.

The ability to change these two arrays during program execution provides tremendous flexibility in modeling the war in each theater. The user can create absolute havoc with the program by making errors on these entries since no automated error checking is programmed. Exercise extreme caution when using this powerful feature.

Methodology, EVENT (5): Section five, EVENT-5, causes ALLOC-2 to try to launch missions when appropriate. SLAM sets XX(274) to 1.0 and XX(268) to 1.0 before calling EVENT-5 so that EVENT-5 knows more launches are possible and ALLOC-2 begins with the top mission request in file 2.

EVENT-5 also takes action only if mission requests are waiting in file 2. EVENT-5 begins by rescheduling itself in 0.432 seconds (0.00001 days). The attribute assignments for rescheduling EVENT-5 are strictly for identification when SLAM TRACE features are in use. The first attribute records the time the call was rescheduled.

EVENT-5 continues to reschedule itself until no further launches are possible. EVENT-5 assumes that the volume of SOF and CR missions is not constrained by the airfield; therefore, a minimum sortie launch interval need not be modeled. Should SOF and CR increase anticipated sortie rates or consolidate basing with other assets becoming runway or facility constrained, this portion of the model must change.

EVENT-5 causes ALLOC-2 to attempt launching sorties by freeing a dummy resource (SLAM RESOURCE number 23). The number of times EVENT-5 activates ALLOC-2 during the simulation is the final capacity of the dummy resource minus one.

Methodology, EVENT (6): Section 6, EVENT-6, simply enters missions requests into the SLAM network at SLAM label MSN. These requests go straight to the mission request queue without delay.

Methodology, EVENT (7): Section 7, EVENT-7, sets up appropriate follow-on missions after considering the outcome of the current sortie. EVENT-7 begins by extracting the critical items identifying the mission from array ATRIB overwriting array A. EVENT-7 also records SOF missions

performed by the Army. Army missions carry a 99.0 in ATRIB(10) for the primary aircraft type.

EVENT-7 records the days between the sortie request and the scheduled departure for successful infils, exfils, and resupplies in SLAM STATs 16-18; likewise, the ranges for successful infils, exfils, and resupplies in SLAM STATs 19-21.

For successful infil missions, EVENT-7 sets up the exfil mission request using USERF-13 to decide if the mission belongs to the Army or Air Force. EVENT-7 uses a random number from stream 6 to decide if the exfil is necessary according to the Army, EVENT-7 assumes the mission is successful and schedules an EVENT-7 call for the day of the exfil. For Air Force exfils within the maximum exfil radius, EVENT-7 schedules the exfil request to enter the SLAM network via EVENT-6 on the day of the desired exfil. For Air Force assigned exfils beyond the exfil capabilities of the defined aircraft, EVENT-7 collects the range in SLAM STAT number 26, but does not schedule the mission.

For successful infil missions, EVENT-7 assumes Army resupply missions are successful and schedules an EVENT-7 call for the day of the resupply. For Air Force resupplies, EVENT-7 assumes that the Air Force can resupply a team that is infilled. EVENT-7 schedules the resupply mission request to enter the SLAM network via EVENT-6 on the day of the desired resupply.

For successful exfil missions, EVENT-7 schedules freeing the team via EVENT-3 after the appropriate delay for the mission region. EVENT-7 places a 3 in attribute A(11) to prevent the scheduled team freeing event from looking like a mission request. All unmet mission requests still do not have a primary aircraft assigned to them so their attribute 11 is 0.0. Since Army exfil missions are not flowing through the SLAM network, but merely simulated with subroutine calls, EVENT-7 must also schedule EVENT-8 to deconflict potential resupply sorties for the same team.

For successful resupply missions, EVENT-7 schedules the next resupply at the appropriate interval for the mission region regardless of any scheduled exfil. Again, EVENT-7 assumes Army resupplies are successful and schedules an EVENT-7 call for the day of the resupply. For Army missions, EVENT-7 schedules an EVENT-8 call to deconflict with the exfil request. EVENT-7 schedules any Air Force resupply mission requests to enter the SLAM network via EVENT-6 on the day of the desired resupply.

For successful combat rescue missions, EVENT-7 collects the scheduling delay in SLAM STAT number 2 and the radius in SLAM STAT number 5. If a team is onboard (ATRIB(9) greater than 0.0), EVENT-7 schedules EVENT-3 to free the team at the turn interval specified for the region in array XX. If rescued SOF or CR crews are onboard, EVENT-7 also schedules EVENT-15 to return them to flying duties after crew rest and mission preparation time elapse.

For unsuccessful rescue missions in the low threat defensive counter air region, EVENT-7 tries again after the minimal CR mission prep time.

After an unsuccessful mission to higher threat regions or unsuccessful SOF missions, EVENT-7 reschedules the mission at the beginning of tomorrow's flying window via EVENT-6.

The only remaining group is an unsuccessful mission that kills the team. If the mission does not fit into any of the previous categories, EVENT-7 schedules EVENT-3 to kill the team as soon as the crash occurs.

Methodology, EVENT (8): Section 8, EVENT-8, prevents scheduling both an exfil and resupply for the same team on the same day. EVENT-8 has no effect on other type missions.

EVENT-8 uses the SLAM routine NFIND to find the mate to the current mission. The resupply and exfil notes same mission number since they support the same team. If the same mission number show up in the mission queue (file 2), it has to be the mate to the current mission. If the mission number shows up on the calendar. EVENT-8 looks at the mission type and number of primary aircraft assigned to determine if it found the mate to the current mission. The resupply is a type 3 mission and the exfil a type 2 mission. Mission requests have no primary aircraft assigned while EVENT calls associated with launched sorties do. EVENT-8 uses SLAM pointers to the data in the files to avoid having to recopy the array ATRIB for each entry being examined.

If the current mission is an unsuccessful exfil and the resupply scheduled today may still be needed, it is simply postponed until tomorrow; otherwise, the resupply is canceled. The user can use a positive XX(269) to tell EVENT-8 to cancel all resupply requests for the team once the exfil request arrives.

If the current mission is a resupply mission, the exfil request for the team would be postponed until at least tomorrow.

Methodology, EVENT (9): Section 9, EVENT-9, sets the system up for the days flying. When the user sets XX(269) positive, EVENT-9 begins by eliminating resupply requests whose exfil request has arrived. Since all mission requests for today are in the queue by the time SLAM executes EVENT-9, the routine only searches the mission requests in file 2 for resupply/exfil mates. The code uses SLAM pointers for efficiency starting from the top of file 2 stopping at each exfil request and looking for a matching resupply. If found, EVENT-9 throws away the resupply request by unlinking it from the file.

When XX(269) equals 0.0, EVENT-9 leaves both the exfil and resupply requests in the queue allowing ALLOC-2 to fly the highest priority mission for which it has the resources. EVENT-8 keeps the model from flying both the resupply and exfil for the same team the same day.

Next, EVENT-9 counts the infil missions waiting for teams in file 1 unless directed by the user to ignore them with a 0.0 in XX(345). The user might wish to exercise this option if the region was team constrained and had plenty of aircraft to better portray the demand actually waiting for aircraft and crews only.

EVENT-9 counts all missions in file 2 awaiting crews and airplanes. Unless the user set XX(269) positive to prevent simultaneous request for exfil and resupply, the total mission demand exceeds that actually needed for today. A positive XX(269) is reasonable if exfils are accomplished when needed. Long delays on the exfil request without intervening resupply missions makes the assumption that the team survives without compromise or capture more tenuous.

EVENT-9 reviews each defined aircraft to determine the number of mission capable planes available to fly today. EVENT-9 pulls a random number from stream 8 for each available aircraft to determine if it is mission capable for today. It also pulls a number from stream 8 to determine which base to direct ALLOC-3 to start looking for the aircraft to seize for maintenance. EVENT-9 builds the maintenance requirement for each type airplane in array B and files the information in file 4 for use by the allocation code. Because SLAM does not do the actual filing until the end of EVENT-9, ALLOC-3 causes no recursive call to subroutine EVENT.

EVENT-9 records the mission capable aircraft and available crews in array XX. Later, ALLOC-2 assumes that mission capable aircraft do launch without ground aborts. EVENT-9 locates the SLAM pointer to resource capacity from information in array LLRSC. The pointer to the capacity of any SLAM resource is one more than the value in LLRSC(RESOURCE NUMBER). EVENT-9 records the capacities in array XX. Aircraft are either mission capable and available to fly, nonmission capable with maintenance requirement specified by file 3 entries, or already in use (NBRUTE). EVENT-9 also counts teams available for use today.

<u>Methodology, EVENT (10)</u>: Section 10, EVENT-10, summarizes the results of today's flying activities.

EVENT-10 begins by recording tankers scheduled for days specified in XX(341) to XX(342) and from XX(342) to the end of the simulation. When values are entered in these XX locations subroutine OTPUT will report tankers scheduled in the final report.

EVENT-10 next checks each aircraft type for resource slack or excessive UTE rate. EVENT-10 defines resource slack as unused generation capability, which EVENT-10 approximates by subtracting the scheduled sorties from available crews or aircraft -- whichever is smaller. A zero slack value shows that type aircraft scheduled enough sorties to use each of its limiting resource at least once. A negative slack value shows that resources are turning during the day so the number of sorties flow actually exceeds the limiting resource. The

slack calculations provide a quick reference to locate the limiting resource for each aircraft type. The actual sortie limitation is much more complex depending upon mission ranges, priorities, flying window length, and number of waves permitted as well as resource limitations.

If the aircraft type has a positive resource capacity at the end of the day (some are left in theater), EVENT-10 records nonpositive slack observations in SLAM STAT number 43 and does UTE rate calculations. If the actual flying hours accumulated today for a type aircraft exceeds the authorized average daily UTE rate, EVENT-10 removes one airplane for the number of days it would take to legally fly the overage, simulating long-term maintenance via file 3 and ALLOC-3. The model thus allows the fleet to spike as high as turn times, resources, mission demand, flying windows, and the other factors permit; however, removing one aircraft a day limits the small fleet sizes associated with SOF and CR to a short surge. If the fleet sizes change, this approximation may no longer be adequate to force the aircraft to stay close to the authorized UTE rate.

EVENT-10 records the total required, scheduled, and successful Air Force SOF missions today in XX(346-348). SOF missions include the Air Force infil, exfil, and resupply missions. Unmet demand for SOF today goes in XX(99) and unmet CR demand in XX(100).

Finally, EVENT-10 prints the values from array XX requested by the user with SLAM RECORD and VARIABLE statements. To avoid recompiling the FORTRAN code every time the user wants different output, the desired plots are listed in array XX(391-400). Basic SLAM permits a maximum of *10 RECORD statements.* The RECORD statement numbers and the entries in XX(391-400) must match and be between 1.0 and 10.0.

EVENT-10 resets the daily observations in array XX(101-250) to 0.0, as required by the number of aircraft types.

EVENT-10 also computes the losses for evading aircrews who are captured today. The predicted time of capture is in attribute 20 and the time of the mission request in attribute 2. EVENT-10 uses pointers to locate the rescue missions and compare the capture times to the current simulation time. For captured crews, EVENT-10 records the days to capture via SLAM STAT number 3 and the combat radius of the captured crew via SLAM STAT number 6. After the statistics, EVENT-10 unlinks mission requests for the captured crews. EVENT-10 begins with CR mission requests waiting in file 2 for aircraft and then reviews the CR mission requests on the SLAM calendar.

EVENT-10 concludes by updating the time to close the flying window tomorrow. EVENT-10 negates the value to prevent ALLOC-2 from starting tomorrow's sorties prematurely.

Methodology, EVENT (11) Section 11, EVENT-11, creates additional airplanes or crews needed by ALLOC-2. ALLOC-2 places the new tanker aircraft requirement in NEWTKE and the tanker SLAM resource number in NTKRSE, the new primary aircraft requirement in NEWACE and the primary

aircraft SLAM resource number in NACRSE, the new primary crew requirement in NPCREW and the primary crew SLAM resource number in NPCRSE, and the new tanker crew requirement in NTCREW and the tanker crew SLAM resource number in NTCRSE. The base for the newly created primary aircraft and crews is encoded in ATRIB(11), and the number and bases for tanker aircraft and crews is in AUXAT.

EVENT-11 uses the SLAM ALTER routine to increase resource capabilities as requested by the values above from COMMON UCOM4.

Methodology, EVENT (12): Section 12, EVENT-12, adjusts the number of resources available in theater as requested by the user via the ENTRY input file. EVENT-12 begins by saving XX(276) in REMXX and the information from the ENTRY file attributes in a local array C. Saving the entries avoids SLAM overwriting the information with any of its routines.

For resource increases, EVENT-12 simply uses the SLAM ALTER routine to increase the aircraft type in attribute 1 by the amount in attribute 2, and the crew type by the amount in attribute 3. It assigns the new resources to the base indicated in attribute 5.

For resource decreases, the current number of resources available at the desired base is the maximum permitted reduction. If further reduction is necessary, EVENT-12 sets IFIL to 1 to indicate the need to further reduce the resources as they become available. EVENT-12 places the unmet reduction in file 4 which has the highest priority for resources. Finally, EVENT-12 restores XX(276) to its original value.

Methodology, EVENT (13): Section 13, EVENT-13, frees primary aircrews. Total crews assigned is in attribute 22, dead crews in attribute 23. If any crewmembers crashed, EVENT-13 turns ALLOC-2 off via XX(276) long enough to free the downed crewmembers and decrease the resource capacity. This reflects the assumption that the crew is essentially dead for sortie generation purposes until rescued and rested. EVENT-15 restores rescued crewmembers. Surviving crewmembers being freed are available for immediate use. Attribute 11 contains the home base for the aircrews.

Methodology, EVENT (14): Section 14, EVENT-14, frees the tanker aircrews. The logic parallels that of EVENT-13 for primary aircrews merely changing to tanker crew attributes. Total tanker crews assigned is in attribute 24, dead tanker crews in attribute 25. The auxiliary attribute array contains the bases for the specific aircraft and aircrews lost.

Methodology, EVENT (15): Section 15, EVENT-15, returns rescued crews to flying duties. Attribute 26 carries the type aircraft flown and attribute 27 the number of rescued aircrews and their home base. The routine assumes that all rescued aircrews can still fly under war conditions. EVENT-15 uses the SLAM ALTER routine to increase the crew resource capacity.

SUBROUTINE INTLC

Subroutine INTLC reads the input files SOFXX, SOFTG, SOFWX, SOFBS, SOFAC, and SOFEN placing the data into the appropriate arrays and providing an echo printout of the data inputs unless suppressed by the user in file SOFXX. INTLC converts percentages to decimal numbers for comparison with 0-1 random numbers generated within the program. INTLC rewinds the file tapes in case multiple runs are needed. After reading the data, INTLC also adjusts the SLAM resource levels to match the team, aircraft, and crew input data. INTLC assigns tanker limits from the SOFXX file to mnemonic variables for the maximum number of A/Rs per sortie (NARLMT) and maximum number of tankers per sortie (NTKLMT). INTLC insures the aircraft and crew allocation code (ALLOC(2, IFLAG)) is off when the simulation begins by setting the time for the end of today's flying window negative. INTLC changes the SOF and CR days for each mission generation rate phase to the simulation times that the phase changes occur. INTLC also converts days aircraft are permitted to surge into the simulation time for the change in hours per day allowed. The model differentiates tanker, CR, and SOF aircraft by the primary mission of each and adjusts the surge rates accordingly. INTLC sets the simulation time that CR can begin flying into each threat type as instructed by file SOFXX. INTLC changes the number of flying waves desired each day into the time interval between attempting them insuring that launches are possible at least once a day. INTLC sets XX(271) to 1.0 if any resource is in the requirements mode. The requirements mode means that the model can make more of the resource whenever necessary.

Finally, INTLC sets XX(276) to 1.0 to indicate that the first flying window may begin on schedule.

Input Parameters: None. Input data resides in files with prefixes SOFXX, SOFTG, SOFWX, SOFBS, SOFAC, and SOFEN. Different versions of the files have additional characters added just before the suffix.

Output Parameters: None. Output goes to the disk print file for data echo or to the appropriate value in the common statement for use by the rest of the model.

Commons: INTLC contains COMMON statements SCOM1, GCOM1, UCOM0, UCOM1, UCOM2, and UCOM3. SCOM1 receives file SOFXX as array XX. INTLC reads the file SOFTG into UCOM1 as array TLOCALE. The basing array BASES is filled from SOFBS. The characteristics of each aircraft defined in file SOFAC go into array ACDATA in UCOM1. INTLC places the maximum possible combat radius for each type mission based upon capabilities of defined aircraft into array CRMAX of UCOM1. Weather data for home stations and regions of interest in file SOFWX goes into array WXDATA in UCOM2. INTLC assigns limits for the number of A/Rs per sortie (NARLMT) and number of tankers per sortie (NTKLMT) in UCOM3. INTLC uses NTKLMT to partition the auxiliary attribute storage array AUXF in UCOM0.

Methodology: INTLC reads file SOFXX first, since data values may suppress the echo printout of input data. It then reads SOFTG, SOFWX, and SOFBS. INTLC reads file SOFAC last after file SOFXX assigns the number of usable aircraft (XX(89)). INTLC bases the maximum combat radius for each mission type on the length of the flying window, the speed of the aircraft, and the combat radius of the aircraft extended by A/R, if applicable. The value in CRMAX reflects the most capable aircraft for each mission. All write statements and formats reflect FORTRAN 77 conventions.

SUBROUTINE LASTAR

Subroutine LASTAR determines the portion of the current refueling schedule in array AR1 accomplished within distance CKDIST along the scheduled route. LASTAR accumulates the distance flown by the primary aircraft from home to the last fuel topoff in DISFLN and sets the number of successful A/R's in ILST. If no refueling occurred since leaving home, DISFLN and A/R are 0.0.

Input Parameters: None.

Output Parameters: None.

Commons: LASTAR contains COMMON statements SCOM1 and UCOM3. UCOM3 supplies the distance to the problem (CKDIST) and the current A/R schedule (array AR1 for NAR1 A/R's). UCOM3 also returns the distance flown by the primary from home to last topoff (DISFLN) and the index of the last successful A/R (ILST).

Methodology: The first column of array AR1 contains the distance flown by the receiver since his last topoff. LASTAR adds these distances until the total exceeds CKDIST then backs off one refueling. The routine assumes all went well until the problem occurred. Bear in mind that AR1 is the current schedule which may already differ from the original schedule.

SUBROUTINE LOSSES

Subroutine LOSSES determines if a primary or tanker aircraft crashes during the mission. IF a crash occurs, LOSSES locates the crash site, adjusts the actual mission information, and asses the status of the team and primary mission. LOSSES explores the possibility of a rescue mission when crashes occur.

Input Parameters: None.

Output Parameters: None.

Commons: LOSSES includes COMMON statements SCOM1, UCOM0, UCOM1, and UCOM3. SCOM1 supplies the current mission data in array ATRIB. UCOM0 contains the tanker home base data, UCOM1 supplies the aircraft characteristics in array ACDATA, and UCOM3 supplies the refueling data. UCOM3 also carries the crash data compiled by LOSSES.

Methodology Begins by assuming no crashes occur and initializes variables: the team is alive (ATRIB(15) equals 0.0); no primary aircraft crashes (NCRSH1 equals 0); a tanker does not cause the mission to abort (TABORT equals 0); the primary aircraft gets home (CRSH1 equals twice the turnpoint distance); likewise, the tanker gets home (CRSH2 equals CRSH1); no unresolved problem remains in the current schedule (CKDIST equals CRSH1); thus, the model already revised the schedule as required for mechanical or weather aborts; the aircraft do not need rescue (CRSHCR= 0.0). After these initial assumptions, LOSSES assigns the mission type to MSN, the aircraft type to NACTYP, the true airspeed to ACTAS, and the number of aircraft to NAC. Having completed initializing variables, LOSSES turns to the random number streams using stream 5 for rescue, stream 1 for SOF, and stream 9 for tankers. LOSSES uses a pair of random numbers for each aircraft with the first determining whether a crash occurs and the second locating the crash site.

LOSSES uses the random numbers and the attrition data in ACDATA to determine whether the primary aircraft crashes. A crash causes as update of the number of primary aircraft crashing (NCRSH1), the site of the primary aircraft crash (CRSH1), and the problem location (CKDIST). If the mission includes tanker support, LOSSES uses routines LASTAR and NEWAR to update the A/R data before considering the possibility of tanker crashes. This reflects the assumption that communications are available, and only required tankers launch.

Each tanker used (NTKUSD) has the potential of crashing. If a tanker crashes, LOSSES locates the first A/R (IFST) assigned to the crashing tanker and the first A/R assigned to the next tanker (INXT). If the crashing tanker is the last tanker, INXT is one higher than the last A/R so that the difference between INXT and IFST is the number of A/R's assigned to the crashing tanker. The model assumes equal probability of the crash occurring during any A/R leg or the leg home. NARLEG represents the number of A/R's accomplished by the crashing

tanker. LOSSES puts tanker crash data into AUXAT. Since the crash could easily occur at low-level without the opportunity for communication, LOSSES assumes the primary aircraft does not discover the problem until the tanker fails to show up for the next A/R (IEND). LOSSES revises the actual hours flown for the crashing tanker column 8 of AR1, locates the crash point for the rescue attempt in CRSHCR, and uses NEWAR to revise the refueling data for the rest of the mission. A tanker crashing after completing all its A/R's does not affect the primary mission or require recomputing A/R support. LOSSES uses MAKECR to create a rescue mission for the tanker crew, if appropriate, before considering the next tanker used. After considering all tankers used, LOSSES updates the actual tanker hours flown and the actual offloads.

LOSSES marks any mission turning short of the original objective as unsuccessful (ATRIB (14) set to 1.0). LOSSES considers the impact of crashes on the primary aircraft, the mission, and the team. LOSSES assumes that the primary aircraft had to fly far enough to reach the predicted crash site to actually crash; therefore, a primary mission aborting for a tanker crash may escape a potential crash himself. If the primary aircraft crashes, the team must be on board to also perish. Mission success up to the time of the crash (ATRIB (14)) and the relationship of the crash site to the objective determine the team status for each type mission. For example, an infil mission which aborted or crashed before the objective still has the team on board. LOSSES sets ATRIB (15) to 1.0 if the team dies in the crash. Finally, LOSSES uses MAKECR to create a rescue mission for the primary aircraft, if appropriate.

SUBROUTINE MAKECR(CRSHLAT,CRSHLON,NHOME,IACFT,NKILL)

Subroutine MAKECR determines whether the CR or SOF crew survives the crash discovered by subroutine LOSSES. If the crew does survive, MAKECR builds the appropriate mission request.

Input Parameters: All calling parameters are input parameters. CRSHLAT and CRSHLON are the coordinates of the crash site. NHOME is the home base of the downed aircraft. IACFT is 1 for a primary aircraft crash and 2 for a tanker crash. NKILL is the number of CR or SOF aircrews flying the airplane that crashed.

Output Parameters: None. Required rescue sortie requests are output using local array A and the SLAM SCHLD routine.

Commons: MAKECR uses COMMON statements SCOM1, UCOM0, and UCOM1. SCOM1 supplies the mission data in array ATRIB. UCOM1 supplies the aircrew survival probabilities in array ACDATA.

Methodology: MAKECR uses SLAM STAT number 45 to record the type of aircraft crashing. MAKECR marks the crew status as lost on the crashing mission's attributes to prevent the model from reusing the crews (downed primary crews in ATRIB(23); downed tanker crews in ATRIB(25)). MAKECR pulls a 0-1 random number from stream 10 to compare with the probability of surviving a crash in ACDATA to decide if a rescue mission is needed.

MAKECR builds the CR mission request in array A. The rescue mission assumes the region (A(1)) and threat (A(4)) of the crashing aircraft. The time of the request is now (A(2)). The coordinates of the crash are placed in A(29) and A(30). The mission type is combat rescue (A(5)). MAKECR obtains the time of anticipated capture (A(20)) from USERF(9). The priority of the mission request for aircraft and crew resources (A(21)) is the sum of the combat rescue mission priority as input via the XX array and the regional priority from TLOCALE.

MAKECR then increments the CR/SOF mission counter to obtain a new mission number (A(6)). If the crash happened to be a rescue mission which had already made the pickup, MAKECR requests a new mission for the original downed crew. In fact, the new rescue mission would attempt to pick up both crews; however, the model limits downed aircrew information to two attributes so the original crew would disappear unless fragged separately. MAKECR uses the SLAM SCHDL routine to schedule subroutine EVENT(6) which enters the mission request into the resource queue.

If the aircrew survives the crash, MAKECR assumes the team also survives if on board and also needs rescue. A(9) carries the number of teams at the rescue site.

MAKECR puts type of aircraft crashing in A(26) and the number and home base of crews in A(27). Successful rescues use this information to return the crews to flying duties at the appropriate bases.

SUBROUTINE NEWAR

Subroutine NEWAR revises the refueling data for aborting missions in section 1, crash of the primary aircraft in section 2, or crash of the tanker aircraft in section 3. The goal of this routine is to get the remaining aircraft home. NEWAR does not attempt to find alternative ways to accomplish the mission with the current assets.

Input Parameters: None. The necessary input data resides in the COMMON statements.

Output Parameters: None. The necessary output updates COMMON statement UCOM3.

Commons: NEWAR contains SCOM1, UCOM0, UCOM1, and UCOM3. UCOM0 contains the tanker basing data in AUXAT. UCOM3 supplies the current A/R schedule in the first NAR1 rows of AR1. UCOM3 also supplies the last A/R completed by the receiver in ILST, the distance flown by the primary aircraft since last topped off in DISFLOWN and the section of NEWAR to execute in ICAUSE.

Methodology: NEWAR is very similar to routines ARSPOT and ARTANK. In fact, some sections are exact copies of portions of the other routines. NEWAR knows which aircraft have been selected, but must revise the previous schedule due to an air abort or crash. NEWAR begins by initializing the same mnemonic variables used and explained in the other routines. NEWAR uses ARTRK to store the receiver's required A/R track in nautical miles. NEWAR branches to the appropriate subsection for the revision code matching the abort cause in ICAUSE.

Methodology, ICAUSE=1, Weather or Mechanical Aborts NEWAR assumes aborts return to the launching base so the distance home (DISTHM) is the distance from home that the abort problem occurs (CKDIST). It computes the actual location of the abort and places its coordinates in ABTLAT and ABTLON. By definition, aborts only occur en route to the objective; aircraft reaching the objective already turn toward home. Since the model assumes the launching base is close to the FLOT, aircraft experiencing mechanical or weather problems returning continue rather than divert. Changing this assumption requires major model modifications adding divert bases.

The distance past the last topoff or A/R for the receiver goes in PASTAR. The number of the next scheduled A/R goes in INXT; routine LASTAR puts the last A/R actually accomplished in ILST. NEWAR places the coordinates of the next scheduled A/R in ARLAT and ARLON. NEWAR iterates IOKAR to indicate the last A/R in array AR1 that reflects the way the mission is actually flown (the last "OK" A/R). When the routine begins, IOKAR is the same as ILST since no changes are needed for A/Rs already accomplished. NEWAR uses NARFLG to remember the last A/R accomplished as planned.

NEWAR assumes that no further refueling occurs if the receiver can

get home from the abort point. If the receiver completed any A/Rs before aborting, NEWAR updates the number of tanker aircraft actually used (NTKUSD) and quits if the receiver's total distance from last topoff to home is less than twice the receiver aircraft's combat radius since no further A/R is necessary. NEWAR does update the last tanker's actual flying hours in column 8 of array AR1. Since NEWAR increments the number of tankers actually used (NTKUSD), it stores the last tanker used on the original schedule in NTKLST. NEWAR assumes communications for aborting aircraft; therefore, the tanker max distance from home in AR1, column 3 is proportional to the primary aircraft distance flown on the current leg before the abort. NEWAR recognizes that abort of less that the receiver combat radius from home require no A/Rs.

NEWAR assumes that the tanker scheduled for the next A/R still performs the A/R but the location changes due to the abort. NEWAR must rendezvous the aircraft to obtain the location of the next A/R. NEWAR begins by calculating the current location of the tanker scheduled for the next A/R (CURTKLAT and CURTKLON). The receiver location is the abort point. NEWAR assumes the tanker and receiver turn toward each other immediately, so the closure speed is the sum of their true airspeeds.

Knowing the aircraft starting locations and closure speed, NEWAR can now calculate the hours until the rendezvous, the location of the rendezvous, and fuel burned between last topoff and the beginning of the next onload. NEWAR assumes the receiver needs 15 minutes of fuel (0.25 hours) to effect the emergency hook-up from the point of rendezvous to start receiving fuel. In other words, the receiver is already flying low-level and cannot glide long enough to get any gas and restart the engines if he is truly completely out of fuel including all his reserves.

Non-VTOL aircraft crash when they run out of gas before effecting the rendezvous. NEWAR assumes VTOL aircraft find a place to set down and shut off the engines to conserve fuel until the tanker can get to them.

At this point, NEWAR knows the rendezvous is within the aircraft capabilities and begins to recalculate AR1. NEWAR first builds receiver requirements in columns 1-3, 10, and 11 as in ARSPOT. The final onload is adjusted down from a complete topoff to the fuel needed to get home with reserves.

Next, NEWAR reschedules the tanker requirements in columns 4-7 similar to code in ARTANK, except that NEWAR must be able to continue a tanker that has already begun refuelings where ARTANK always starts the tanker from its first A/R.

For tanker hours in column 8, NEWAR recognizes when tanker hours are expended even though no A/R occurs due to the abort. NEWAR recomputes the hours for the tanker even when the same number of A/Rs occur because the location of the A/R may change changing the hours flown.

For loiter adjustments for the tanker, NEWAR assumes the tanker tracks keep the tanker roughly parallel to the receiver's progress on the primary route. Thus, tanker progress between A/Rs is at the receiver airspeed, whereas tanker progress before the first or after the last A/R is at the tanker's airspeed. NEWAR does not recompute crew requirements.

NEWAR computes hours flown by any tankers launched in support of the aborting sortie, but not used. Since the old schedule remains in AR1, for rows following the recomputed row IOKAR through the original number of A/Rs NAR1, NEWAR looks for tanker numbers differing from the number of the last tanker NEWAR used. To decide if scheduled tankers launched, NEWAR calculates the distance the receiver would have flown from the last accomplished A/R to the A/R scheduled for the next tanker and converts the distance to hours using the receiver's airspeed. If the receiver's flying time is less than the time for the tanker to get to the rendezvous at tanker true airspeed, the tanker is already airborne and has expended flying hours.

NEWAR totals the pounds of fuel actually offloaded to the receiver in SHORT and updates attribute 10. NEWAR also totals all tanker aircraft flying hours supporting this sortie in TKHRS and updates attribute 7. Finally, the new schedule becomes the established schedule in case the sortie has other problems so NEWAR updates the total A/Rs scheduled (NAR1) to the last A/R NEWAR requires (ILST). AR1 now reflects actual refueling information considering the weather or mechanical abort for use by the rest of the model.

Methodology, ICAUSE=2, Primary Aircraft Crashes: NEWAR with ICAUSE=2 is appropriate when the primary aircraft crashes or can no longer receive fuel via A/R. Currently, the model uses this section only for crash of the primary aircraft in LOSSES. NEWAR assumes the crash may be so sudden that the receiver does not have an opportunity to communicate with the tanker; therefore, the tanker discovers the problem when the receiver fails to show up for the next A/R. Since the receiver does not show, NEWAR drops the offload at the next A/R and revises the tanker's hours. NEWAR assumes this tanker notifies home station of the receiver no-show thus stopping further tanker launches in support of this sortie. NEWAR computes hours flown by any tankers launched in support of the aborting sortie, but not used as discussed with ICAUSE=1.

Methodology, ICAUSE=3, Tanker Crashes: NEWAR with ICAUSE=3 is appropriate when the tanker crashes before finishing the assigned A/Rs. NEWAR assumes the tanker crash may be sudden precluding communication with the receiver; therefore, the receiver discovers the problem when the tanker fails to show for the next A/R.

NEWAR assumes the receiver aborts and notifies home station of the tanker no-show. How serious the problem is depends upon the receiver's distance from home and his fuel status. If the receiver can recover without refueling, NEWAR simply sets the last A/R (NAR1) to the last A/R already accomplished (ILST) and sends the receiver home.

When the receiver needs more fuel, NEWAR locates the next scheduled tanker relative to home station, as explained above with ICAUSE=1; however, the ICAUSE=3 abort occurs at a scheduled A/R location. If no more tankers are scheduled, NEWAR will use SPARE to get the next tanker.

NEWAR must compute a rendezvous for the tanker and receiver aircraft. NEWAR assumes the seriousness of the situation causes the aircraft to immediately turn toward each other so that the closure speed is the sum of the true airspeeds. NEWAR requires the receiver to have 15 minutes of fuel to effect the hook-up. As above, NEWAR assumes fixed wing aircraft without sufficient fuel to rendezvous crash while VTOL-capable aircraft land and shut down, if necessary, to avoid burning fuel. NEWAR crashes the aircraft only after all reserves are exhausted; therefore, missing an A/R very close to home may leave the receiver enough fuel to limp home, but not enough to effect a rendezvous although the possibility is extremely remote.

Computing the receiver requirements after the emergency rendezvous parallels previous logic iterating A/Rs (INXT) until completing the A/R which lets the receiver reach home with reserves (NAR1). NEWAR adjusts the last receiver onload to the minimum requirement.

NEWAR begins revising the tanker schedule after the last accomplished A/R (ILST). NEWAR iterates IOKAR to show the last revised A/R for AR1. NEWAR assumes the sortie can use all tankers assigned to this mission with the exception of the one that crashed; any additional tanker support must come from subroutine SPARE. After revising the A/R schedule (AR1), the actual number of A/Rs (NAR1), and the actual number of tankers used (NTKUSD) for the mission, NEWAR quits.

Code after label 999 suppresses the old information in the AR1 array and prints the new schedule. This code is very useful in model development, so it is left for use in verifying future modifications to the routine. Currently, NEWAR does not execute this code.

SUBROUTINE OTPUT

Subroutine OTPUT is the last subroutine executed by the program on each run, thus providing the place to extract any desired information before the program stops. If XX(341) is greater than 0.0, the routine writes the average number of tankers scheduled for the number of days specified in XX(341) through XX(342). The total number of tanker missions during these days is in XX(343). Likewise, the routine writes the average number of tankers scheduled in the time period from XX(342) to the end of the simulation. The total number of tanker missions during these days is in XX(344). If an error condition exists within SLAM causing early termination of the run, OTPUT also dumps the contents of all SLAM files to assist in identifying the problem. Additional code may be added to write statistics to a special output file if desired.

Input Parameters: None.

Output Parameters: None. Output goes to both the SLAM output file and to the special output file.

Commons: OTPUT contains COMMON statement SCOM1.

Methodology: OTPUT contains write statements as discussed above. OTPUT uses the SLAM PRNTF routine to dump the files on error stops.

SUBROUTINE PLANE(NAC1,NAC2,NPRI1,NPRI2,TYPMSN,NREQ,NCRREQ,THREAT,NACTYP, FLYHR,NBASE,NACAVL,NCRAVL,NACRES,NCRRES,MSNPRI,NEWAC,NEWCR,TLAT,TLON)

PLANE is a search routine that examines the specified aircraft in ACDATA over the specified mission priorities looking for the required number of aircraft and aircrews capable of performing the mission within the anticipated threat environment. The search stops when PLANE finds a capable aircraft with aircrews or determines there are no capable aircraft/crews available.

Input Parameters: The input parameters are NAC1, NAC2, NPRI1, NPRI2, TYPMSN, NREQ, THREAT, TLAT, and TLON. TLAT and TLON are the coordinates of the objective of the mission. The search begins with aircraft type NAC1, and continues until PLANE finds a capable aircraft or looks at aircraft type NAC2 without finding an aircraft. NAC2 should always be greater than or equal to NAC1. PLANE starts the search with mission priority NPRI1, looks at the specified aircraft types, and continues to search through mission priority NPRI2. NPRI2 should be greater than or equal to NPRI1. TYPMSN is the mission type (infil=1.0, exfil=2.0, resupply=3.0, rescue=4.0, and tanker=5.0). NREQ is the number of aircraft needed. THREAT is 1.0, 2.0, or 3.0 reflecting the anticipated threat to the aircraft with 1.0 being the most intense threat.

Output Parameters: Output parameters are NCRREQ, NACTYP, FLYHR, NBASE, NACAVL, NCRAVL, NACRES, NCRRES, MSNPRI, NEWAC, and NEWCR. NCRREQ is the number of aircrews required to fly the mission. NACTYP is the type aircraft selected. FLYHR is the number of hours required for the mission. NBASE is the home base for the chosen aircraft. NACAVL is the number of aircraft available to fly. NACRES is the resource number for the chosen aircraft, and NCRRES is the resource number for the the aircrews. MSNPRI is the priority of the mission for the selected aircraft. NEWAC is the number of new aircraft required for the selected aircraft to do the mission, and NEWCR is the number of new crews required.

Commons: PLANE includes COMMON statement UCOM1.

Methodology: PLANE converts the mission priorities in NPRI1 and NPRI2 to indices for ACDATA. PLANE sets the mission priority and starts at aircraft type NAC1 proceeding to type NAC2 until finding a plane that does the mission at that mission priority. When PLANE finds an aircraft that does the mission, it asks whether the plane meets the specified threat and has the required number of aircraft available. It then searches through the bases at which that type aircraft is stationed to find the base with aircraft that is closest to the objective. PLANE computes the time required for the mission (FLYHR) and uses that information to check whether augmentation is required. It places the number of crews required in NCRREQ. If the base under consideration has the required crews available, PLANE accepts the aircraft. It places the base number in NBASE. If no aircraft or crew is available, PLANE zeros all output parameters and returns.

SUBROUTINE SPARE(NTKTYP,NSPARE,NBASE)

SPARE documents the use of a tanker spare. A solid concept of operations governing tanker spare requirements was not available; however, the staff with tanker experience maintained that penetration missions which put the receiver in jeopardy would not launch without a tanker spare available. Therefore, the model optimistically assumes a spare is available and merely documents the request. Flying hours accrued by the spare do count toward UTE rate limits within the model.

Input Parameters: The input parameter NTKTYP specifies the aircraft type of the tanker scheduled for the mission, and NBASE is the base at which the spare is needed.

Output Parameters: The output parameter NSPARE is always 1 since a spare is always considered available.

Commons: SPARE includes COMMON statements SCOM1, UCOM0, UCOM1, and UCOM3.

Methodology: Currently, assume spare available. In the future, recommend developing a SPARE tanker queue and supporting logic.

SUBROUTINE STATS(ISTAT)

STATS is a two-part routine for recording items of interest about the current sorties while the information is still available. STATS records details about infil, exfil, resupply, combat rescue, or tanker missions in array XX, describing both the scheduled and actual mission characteristics. STATS also records details about both the scheduled and actual use of the selected aircraft. The daily information is lost at the end of each day unless the user requests the items using SLAM RECORD and VARIABLE statements in the SOFCD file.

Input Parameters: The input parameter ISTAT is 1 when recording scheduled information and 2 when recording actual information.

Output Parameters: None. STATS records information in array XX.

Commons: STATS uses COMMON statements SCOM1, UCOM0, UCOM1, UCOM2, and UCOM3 to obtain and pass information.

Methodology: STATS computes the XX array index based upon mission type for mission data or aircraft type for aircraft data. STATS then extracts the applicable information from the attributes. Note that tanker missions are support missions so the requirement depends upon the selected primary aircraft; therefore, STATS increments the tanker requirement (XX(141)) by the number of tanker sorties scheduled. ALLOC supplies the unmet tanker requirement. Since only 10 items are available per aircraft, STATS records the number of tanker A/R's (XX(156)) and fuel offload (XX(158)) only if not receiver capable; otherwise, these XX locations contain the number of A/R's and fuel onloaded as a receiver. STATS interprets any tanker mission that passes fuel to a receiver as a successful mission instead of requiring the tanker to finish all scheduled A/R's to be successful. Information on XX variables 94 through 250, as described in the INTRODUCTION, is available from STATS.

SUBROUTINE TARGET(NREGON,TLAT,TLON)

TARGET locates the target latitude and longitude for new CR and SOF missions.

Input Parameters: NREGON is the region for the mission.

Output Parameters: TLAT and TLON are the geographical coordinates of the target location.

Commons: TARGET uses COMMON statements SCOM1 and UCOM1. UCOM1 supplies the target location data in array TLOCALE.

Methodology: TARGET assumes the targets are uniformly distributed throughout the regions. It first draws two random numbers; it uses stream 4 for CR missions and stream 1 for SOF missions. It uses the first random number to choose the target latitude and the second to choose the target longitude.

FUNCTION USERF(IFN)

USERF contains 13 separate functions used by other routines in the model. USERF sets the mission combat radius and threat type, resolves fractional mission requests, picks the region, determines take-off weather delays, checks for weather or mechanical aborts enroute, computes turn time for recovering primary and tanker aircraft, decides whether to launch another wave of sorties today, sets crew rest and mission prep times for the primary and tanker crews, and decides if a SOF mission belongs to the Army or Air Force. FORTRAN does not support recursive function calls so no USERF function calls another USERF function, either directly or indirectly, through other routines. recursive calls cause disastrous execution errors; therefore, avoid recursion in future additions or modifications. USERF replaces the dummy SLAM function USERF.

Input Parameters: IFN is the integer number of the desired function within USERF.

Output Parameters: None. The function assigns the desired value to USERF, which the calling routine treats as a variable.

Commons: USERF contains COMMON statements SCOM1, UCOM0, UCOM1, UCOM2, and UCOM3.

Methodology, USERF(1): USERF(1) passes the region number to subroutine TARGET, which actually computes the target location. USERF(1) places the target latitude in ATRIB(29) and the longitude in ATRIB(30). For more information see the write-up on TARGET.

Methodology, USERF(2): USERF(2) assumes the threat type conforms to definitions used in the SOF Master Plan. USERF(2) assigns the threat type facing the primary aircraft for infil and rescue missions. Exfil and resupply missions retain the threat type associated with the team's infil mission.

The probability of facing a Type I threat lies in XX(301-310) for regions 1-10, and the probability of facing at least a Type II threat lies in XX(311-320). USERF(2) pulls one random number from stream 1 to determine the threat type for infils and stream 4 for CR. The model contains a provision to limit CR launches if the threat is too intense in the beginning of the conflict. The user enters a delay in days from the start of open conflict to the start of CR missions in XX(278-280) for threat Types I-III, respectively. Subroutine INTLC converts the delays to the earliest time permitted for Type I-III threat CR missions.

Methodology, USERF(3): Since the model cannot launch a fractional mission, it must convert non-integral missions rates to an integer. Rounding the request to an integer could lead to significant error over the simulation. For example, rounding 2.4 to 2 understates the mission demand by four missions in only 10 days. The model uses the fractional portion of the mission rate as the probability that the request should round up. In the above example, forty percent of the time the model

would generate three missions instead of only two, resulting in an average of 2.4 missions per day. The model uses random stream 7 to resolve fractional requests.

Methodology, USERF(4): USERF(4) assigns the region number based upon the cumulative probability distribution of missions for regions 1-10 found in TLOCALE(1-10,1) for SOF or TLOCALE(11-20,1) for CR. The model pulls a random number from stream 1 for SOF infils and stream 4 for CR missions. Exfils and resupplies assume the same region as the team's infil.

Methodology, USERF(5) USERF(5) checks for weather delays at take-off. The model pulls three values from stream 5 for CR or stream 2 for SOF missions. The first two are passed to WXCELL to determine the ceiling and visibility values at take-off. The third determines the portion of the flying window covered by the weather delay. For delayed CR sorties, the model passes the hours delayed to SLAM STAT number 9; for delayed SOF sorties, to SLAM STAT number 28. USERF(5) converts the delay from hours to days since the simulation clock runs in days.

Methodology USERF(6): USERF(6) checks launching sorties for weather or mechanical aborts. The routine begins by pulling three random numbers from stream 5 for CR missions or stream 2 for SOF missions. The first random number determines the fraction of the mission flown if WABORT indicates the mission weather aborts. USERF(6) compares the second random number to the primary aircraft mechanical abort rate in ACDATA to determine if a mechanical failure occurs. The third random number determines the fraction of the mission flown for mechanical aborts. If the mission includes tanker aircraft, USERF(6) draws a random number from stream 9 to determine whether the tanker aborts for a mechanical problems. USERF(6) assumes only the first abort point actually occurs, and mission aircraft turn toward home at that point. The model assigns USERF the value 1.0 to show a weather or mechanical abort occurs or 0.0 to show no abort. USERF(6) computes the probability of weather abort on a CR mission with SLAM STAT number 11 and the probability of mechanical abort on a CR mission with SLAM STAT number 12. For SOF missions, SLAM STAT number 30 is the probability of weather abort, and SLAM STAT number 31 is the probability of mechanical abort. For aborting sorties, USERF(6) revises the turnpoint to the abort point and calls routines LASTAR and NEWAR at revise the refueling schedule if the mission uses tankers.

Methodology, USERF(7): USERF(7) converts the turn time required for the primary aircraft from hours shown in ACDATA to days for use in the model. If an airplane is mission capable in the morning, the model assumes the plane remains flyable for the day requiring only a delay for servicing or light maintenance between sorties.

Methodology, USERF(8): USERF(8) converts the turn time required for the tanker aircraft from hours shown in ACDATA to days for use in the model. As with the primary aircraft, the model assumes flyable aircraft remain flyable requiring only a delay for servicing or light

maintenance between sorties.

Methodology, USERF(9): USERF(9) sets the capture date for downed crewmembers. Since the input CR sortie rates already include a 35% cut for nonsurviving or immediately captured crews, the model assumes all generated CR mission requests are valid at least through the day of the request. The model uses random stream 10 and SLAM's poisson distribution generator to calculate the integral number of days the aircrew evades. The model permits a different mean for each region. A mean value of 2.0 limits causes about 99 percent of the crews to evade three days or less, which reasonably approximates historical data.

Methodology, USERF(10): USERF(10) decides whether the model should attempt to launch more sorties today. The user specified the number of launch waves desired each day in XX(86), which the subroutine INTLC converts to the time in days between launch windows in XX(272). The user also supplied the hours for flying operations in XX(85) which the INTLC converts to the time in days for flying to stop, XX(270). The routine indicates that enough time remains to launch another wave of sorties by assigning USERF the value 1.0; otherwise, the routine uses 0.0 to indicate no more waves fit today.

Methodology, USERF(11): USERF(11) computes crew rest and mission preparation delays for surviving primary aircraft crews. For CR missions, the threat determines the delay. CR crew on missions in Type III threat need only time to prepare for the next mission (XX(384)). CR missions in Type I or II threats require crew rest and a longer mission preparation time (XX(380) and XX(385)). For SOF mission, crews from unsuccessful missions receive only crew rest (XX(380)) before reattempting the mission; otherwise USERF(11) assumes the crew needs both crew rest and mission preparation time before assignment to a new mission. The mission preparation time can vary by mission type for infil, exfil, or resupply missions as shown in XX(381-383). The routine converts the hours shown in the XX array to days for assignment to USERF.

Methodology, USERF(12): USERF(12) computes crew rest and mission preparation delays for tanker crews according to the threat type of the primary mission. Crews supporting low threat missions (Type III) receive crew rest and the minimum CR mission planning time (XX(380)+XX(384)). Tanker crews supporting other missions receive crew rest and the maximum CR planning time (XX(380)+XX(385)). The routine converts the hours in the XX array to days before assignment to USERF.

Methodology, USERF(13): USERF(13) determines whether a SOF mission belongs to the Army or the Air Force based upon user inputs in XX(371-375) and a random number from stream 3. The routine assigns USERF the value 99.0 to indicate the Army does the mission or the value 0.0 to indicate the Air Force does the mission.

SUBROUTINE WABORT

WABORT determines the ability of the selected aircraft to meet the weather encountered in the objective area.

Parameters: None.

Commons: WABORT uses COMMON statements SCOM1, UCOM0, UCOM1, and UCOM2 to receive and pass information.

Methodology: WABORT begins by setting indices to the appropriate mission weather minimums in array ACDATA for the primary aircraft and the tanker, if assigned. (Mission weather minimums begin in Row 38 of ACDATA.) Attribute 10 supplies the primary aircraft type, attribute 12, the tanker types. Attribute 1 supplies the mission area, and attribute 5 supplies the mission type.

WABORT pulls two random numbers from stream 5 for CR missions or stream 2 for SOF missions. WABORT passes the first as RNUM1 for determining the appropriate cell. WABORT passes the second to WXCELL as RNUM2 for interpolating the exact value within the cell. Since the different weather phenomena are not actually independent, WABORT uses the same two random numbers for all phenomena for this mission.

WABORT then checks the primary and tanker aircraft weather minimums against the simulated observations for ceiling, visibility, wind, rain, and turbulence. For efficiency, WABORT stops at the first value beyond the aircraft's capabilities.

WABORT sets attribute 14 to 1.0 for weather aborts; otherwise, WABORT sets attribute 14 to 0.0 indicate the mission's success to this point.

NOTE: Since WABORT initializes attribute 14 to indicate initial mission success, WABORT must remain the first routine called within subroutine ALLOC for launching sorties.

SUBROUTINE WXCELL(NROW,NCOL1,NCOL2,RNUM1,RNUM2,VALUE)

WXCELL interpolates between the weather observation points supplied by USAFETAC to produce an exact weather value.

Input Parameters: The first five parameters are input parameters which subroutine WABORT sets before calling WXCELL. NROW is the row in the weather file corresponding to home for take-off observations or to the appropriate region for observations in the objective area (row 2 for home, row 3-12 for areas 1-10). NCOL1 is the first column and NCOL2 the last column in the weather file appropriate to the requested value (probabilities for ceilings in columns 1-6, turbulence in 7, visibilities in 8-11, wind in 12-15, and heavy rain in 16). RNUM1 and RNUM2 are random numbers between 0-1. Recall that subroutine INTLC converts with the 0-1 random numbers.

Output Parameters: VALUE is the weather observation for the point and time requested.

Commons: None.

Methodology: WXCELL compares RNUM1 with the cumulative probabilities in NCOL1 through NCOL2 of row NROW in the weather file. The first column value to equal or exceed RNUM1 identifies the column containing the maximum weather value. Weather values are in row 1 of the weather file. WXCELL uses RNUM2 to interpolate between the maximum value column and the preceding column row 1 entries to obtain the requested weather observation returned as VALUE.

Appendix B. SLAM Code

This appendix contains the SLAM code used in the simulation model. The code begins on the next page.

```
GEN,KRAUS,CRASOFM1,10/15/86,10,Y,N,Y,Y,Y/1,132;  FILE=SOFCD.DAT
LIMITS,5,30,675;
STAT,1,RESCUE SCHED DLY;                         COMBAT RESCUE STATS
STAT,2,DAYS TO RESCUE;
STAT,3,DAYS TO CAPTURE;
STAT,4,ALL CR MSN RANGE;
STAT,5,RANGES RESCUED;
STAT,6,RANGES CAPTURED;
STAT,7,RESCUE TOO LONG;
STAT,8,CR WEATHER CANX;
STAT,9,HRS WEATHER DLY;
STAT,10,CR PROB FAILURE;
STAT,11,CR PROB WX FAIL;
STAT,12,CR PROB MX FAIL;
STAT,13,INFIL SCHED DLY;                         SOF STATS
STAT,14,EXFIL SCHED DLY;
STAT,15,RESUP SCHED DLY;
STAT,16,DAYS TO INFIL;
STAT,17,DAYS TO EXFIL;
STAT,18,DAYS TO RESUP;
STAT,19,USAF INFIL RANGE;
STAT,20,USAF EXFIL RANGE;
STAT,21,USAF RESUP RANGE;
STAT,22,SCH INFILS BY AC,10/1./1.;
STAT,23,SCH EXFILS BY AC,10/1./1.;
STAT,24,SCH RESUPS BY AC,10/1./1.;
STAT,25,SCH RESCUE BY AC,10/1./1.;
STAT,26,SCH REFUEL BY AC,10/1./1.;
STAT,27,SOF WEATHER CNX;
STAT,28,HRS WEATHER DLY;
STAT,29,SOF PROB FAILURE;
STAT,30,SOF PROB WX FAIL;
STAT,31,SOF PROB MX FAIL;
STAT,32,ACFT 1 UTE RATE;                         ACFT/CREW RESOURCE STATS
STAT,33,ACFT 2 UTE RATE;
STAT,34,ACFT 3 UTE RATE;
STAT,35,ACFT 4 UTE RATE;
STAT,36,ACFT 5 UTE RATE;
STAT,37,ACFT 6 UTE RATE;
STAT,38,ACFT 7 UTE RATE;
STAT,39,ACFT 8 UTE RATE;
STAT,40,ACFT 9 UTE RATE;
STAT,41,ACFT 10 UTE RATE;
STAT,42,TOT MSNS BY ACFT,10/1./1.;
STAT,43,ACFT OR CREW LIM,10/1./1.;
STAT,44,TANKER TYPE REQ,2/1./1.;
STAT,45,CRASHES BY TYPE,10/1./1.;
STAT,46,ACFT USE TANK I,10/1./1.;
STAT,47,ACFT USE TANK II,10/1./1.;
STAT,48,ARMY INFIL RANGE,20/0./50.;
STAT,49,ARMY EXFIL RANGE,20/0./50.;
STAT,50,ARMY RESUP RANGE,20/0./50.;
```

```
;
PRIORITY/2,LVF(21);                MSN PRIORITY SET IN XX(1-10),
;                                  REGION PRIORITY SET IN TLOCALE(I,6)
NETWORK;
       RESOURCE/ATEAMS(1),4,1;
       RESOURCE/RTEAMS(1),4,1;
       RESOURCE/T1ACFT(1),4,3,2;
       RESOURCE/T2ACFT(1),4,3,2;
       RESOURCE/T3ACFT(1),4,3,2;
       RESOURCE/T4ACFT(1),4,3,2;
       RESOURCE/T5ACFT(1),4,3,2;
       RESOURCE/T6ACFT(1),4,3,2;
       RESOURCE/T7ACFT(1),4,3,2;
       RESOURCE/T8ACFT(1),4,3,2;
       RESOURCE/T9ACFT(1),4,3,2;
       RESOURCE/T10ACFT(1),4,3,2;
       RESOURCE/T1CREW(1),4,2;
       RESOURCE/T2CREW(1),4,2;
       RESOURCE/T3CREW(1),4,2;
       RESOURCE/T4CREW(1),4,2;
       RESOURCE/T5CREW(1),4,2;
       RESOURCE/T6CREW(1),4,2;
       RESOURCE/T7CREW(1),4,2;
       RESOURCE/T8CREW(1),4,2;
       RESOURCE/T9CREW(1),4,2;
       RESOURCE/T10CREW(1),4,2;
       RESOURCE/DUMMY(1),2;

; THIS CODE IS CRITICAL TO THE FORTRAN/SLAM CALENDAR INTERFACE.
      CREATE;                            COUNT RESOURCES & MISSIONS IN QUE
         ACT,1.,,EDAY;                   SCHEDULE END OF DAY STATISTICS
         ACT,.0002;                      DELAY START UNTIL MSNS IN QUE
STRT  EVENT,9;                           DO AM STATS & SET ACFT AVAILABLE
WAVE  ASSIGN,XX(273)=USERF(10),2;
         ACT,.0000001,,FLY;              DELAY WHILE MAINTENANCE CHECKS ACFT
         ACT,XX(272),XX(273).GE.1.,WAVE; NEED MORE WAVES
         ACT;
      TERM;
FLY   ASSIGN,XX(274)=1.,XX(268)=1.;      SET FLY FLAG TO "ON",
;                                        SEARCH FROM TOP OF FILE 2
      EVENT,5;                           FLY ALL POSSIBLE SORTIES
DONE  TERM;                              NO MORE WAVES TODAY
EDAY  EVENT,10;                          FINISH TODAYS STATS-RESET
         ACT,.0002,,STRT;                START TOMORROW
         ACT,1.,,EDAY;                   SIMULATE 1 DAY

; THIS CODE SCHEDULES START FOR BOTH SOF AND RESCUE MISSIONS
      CREATE,,.0001;
      ASSIGN,XX(12)=XX(12)-1.,XX(282)=XX(282)-1.;  SET STOP TIMES FOR SOF/CR
         ACT,XX(11),,NEW1;               START SOF GENERATION
         ACT,XX(281),,NEW2;              START RESCUE GENERATION
```

```
; THIS CODE GENERATES SOF INFIL MISSIONS
NEW1  ASSIGN,ATRIB(2)=TNOW,2;                SET CREATION TIME
         ACT,,,SOF1;
         ACT,1.,TNOW.LT.XX(12),NEW1;         SCHEDULE NEXT DAY IF APPLICABLE
         ACT;
      TERM;
SOF1  GOON,1;                                SET INFIL RATES
         ACT,,TNOW.GE.XX(15),SOF4;           START SOF PHASE IV
         ACT,,TNOW.GE.XX(14),SOF3;           START SOF PHASE III
         ACT,,TNOW.GE.XX(13),SOF2;           START SOF PHASE II
         ACT,,,BEG1;                         STILL IF SOF PHASE I
SOF4  ASSIGN,XX(16)=XX(19);                  SET SOF PHASE IV RATE
         ACT,,,BEG1;
SOF3  ASSIGN,XX(16)=XX(18);                  SET SOF PHASE III RATE
         ACT,,,BEG1;
SOF2  ASSIGN,XX(16)=XX(17);                  SET SOF PHASE II RATE
BEG1  ASSIGN,ATRIB(5)=1.,ATRIB(9)=1.,
            XX(20)=USERF(3),1;               INFIL MSN FOR ONE TEAM,
;                                            COMPUTES TOTAL INFILS FOR TODAY
         ACT,,XX(20).GT.0.,NXT1;              CONTINUE IF ANY INFILS ARE NEEDED
         ACT;
      TERM;
NXT1  ASSIGN,XX(90)=XX(90)+1.,ATRIB(6)=XX(90),
            ATRIB(1)=USERF(4),ATRIB(29)=USERF(1),
            ATRIB(4)=USERF(2),XX(20)=XX(20)-1.,
            ATRIB(21)=ATRIB(21)+XX(1),2;INCREMENT INFIL/RESCUE MSN COUNTER,
;                                            ASSIGN MSN #,TARGET,THREAT,
;                                            UPDATE TODAY'S INFIL MISSION COUNT,
;                                            SET MSN PRIORITY
         ACT/1,,,QMEN;                       REQ SOF TEAM
         ACT,.0000001,XX(20).GT.0.,NXT1; REQUEST ANOTHER INFIL MSN IF REQ'D
         ACT;
      TERM;
QMEN  AWAIT(1),ALLOC(1);                     TEAM QUE
ARMY  ASSIGN,ATRIB(10)=USERF(13),ATRIB(8)=ATRIB(3)*
            .0003472,1;                      1/(120*24)=.0003472
         ACT,,ATRIB(10).EQ.0.,QUE;
         ACT,ATRIB(8);
ARM7  EVENT,7;
      TERM;

; THIS CODE GENERATES COMBAT RESCUE MISSIONS
NEW2  ASSIGN,ATRIB(2)=TNOW,2;                SET CREATION TIME
         ACT,,,CR1;
         ACT,1.,TNOW.LT.XX(282),NEW2;        SCHEDULE NEXT DAY IF APPLICABLE
         ACT;
      TERM;
CR1   GOON,1;                                SET RESCUE RATES
         ACT,,TNOW.GE.XX(285),CR4;           START RESCUE PHASE IV
         ACT,,TNOW.GE.XX(284),CR3;           START RESCUE PHASE III
         ACT,,TNOW.GE.XX(283),CR2;           START RESCUE PHASE II
         ACT,,,BEG2;                         STILL IN RESCUE PHASE I
```

```
CR4   ASSIGN,XX(286)=XX(289);            SET RESCUE PHASE IV RATE
         ACT,,,BEG2;
CR3   ASSIGN,XX(286)=XX(288);            SET RESCUE PHASE III RATE
         ACT,,,BEG2;
CR2   ASSIGN,XX(286)=XX(287);            SET RESCUE PHASE II RATE
BEG2  ASSIGN,ATRIB(5)=4.,ATRIB(9)=0.,
             XX(290)=USERF(3),1;         RESCUE MSN FOR AIRCREW (NO A-TEAM),
;                                        COMPUTES TOTAL RESCUE MSNSS FOR TODAY
         ACT,,XX(290).GT.0.,NXT2;        START RESCUES IF NEEDED
         ACT;
      TERM;
NXT2  ASSIGN,XX(90)=XX(90)+1.,ATRIB(6)=XX(90),
             ATRIB(1)=USERF(4),ATRIB(29)=USERF(1),
             ATRIB(4)=USERF(2),ATRIB(20)=USERF(9),
             XX(290)=XX(290)-1.,ATRIB(21)=ATRIB(21)+XX(4),2;
;                                        INCREMENT INFIL/RESCUE MSN COUNTER,
;                                        ASSIGN MSN #,TARGET,THREAT,
;                                        SET CAPTURE DATE, UPDATE TODAY'S
;                                        RESCUE MISSION COUNT,SET MSN PRIORITY
         ACT/2,ATRIB(8),,QUE;            REQ CR ACFT
;             SEND RESCUE MSN REQUEST TO ACFT QUE WHEN THREAT PERMITS
         ACT,.0000001,XX(290).GT.0.,NXT2; REQUEST ANOTHER RESCUE MSN IF REQ'D
         ACT;
         ACT;
      TERM;

; FOLLOWING CODE FOR CR & SOF MISSIONS
MSN   ENTER,1;                           MSN REQUEST FOR TODAY
QUE   AWAIT(2),ALLOC(2);                 ACFT QUEUE
SCHD  EVENT,7;                           SCHEDULE FOLLOW UP MISSIONS
;                                        SCHEDULE TEAM RELEASE IF NEEDED
DCON  EVENT,8,1;          PREVENT EXFIL & RESUP FOR SAME TEAM ON SAME DAY
         ACT,ATRIB(19),ATRIB(16).LT.0.,WDLY; DID NOT FLY
         ACT,ATRIB(19);                  DID FLY
GON1  GOON,2;
         ACT,ATRIB(8),,FLYP;             FLY PRIM ACFT
         ACT,ATRIB(16),ATRIB(13).GT.0.,FLYS; FLY TANKERS
         ACT;
      TERM;
FLYP  ASSIGN,ATRIB(8)=USERF(7);          RECYCLE DLY PRIM ACFT
         ACT,ATRIB(8),,FREP;             TURN PRIM ACFT
         ACT;
FLCP  ASSIGN,ATRIB(8)=USERF(11);         RECYCLE DLY PRIM CREW
         ACT,ATRIB(8);                   CREW REST + PREP
FRCP  EVENT,13;                          FREE PRIM CREW
      TERM;
FREP  EVENT,1;                           FREE PRIM ACFT
      TERM;
FLYS  ASSIGN,ATRIB(16)=USERF(8);         RECYCLE DLY SUP'T ACFT
         ACT,ATRIB(16),,FRES;            TURN SUP'T ACFT
         ACT;
FLCS  ASSIGN,ATRIB(16)=USERF(12);        RECYCLE DLY SUP'T CREW
```

```
        ACT,ATRIB(16);                      CREW REST + PREP
FRCS  EVENT,14;                             FREE SUP'T CREW, CLEAR AUXAT
      TERM;
FRES  EVENT,2;                              FREE SUP'T ACFT
      TERM;

WDLY  ASSIGN,ATRIB(8)=XX(380)/24.,4;        TRIED BUT DID NOT FLY
        ACT,,,FREP;                         GO FREE PRIMARY ACFT
        ACT,ATRIB(8),,FRCP;                 MIN CREW REST FOR PRIM CREW
        ACT,,ATRIB(13).GT.0.,FRES;          GO FREE TANKER IN REQ'D
        ACT,ATRIB(8),ATRIB(13).GT.0.,FRCS; MIN CREW REST FOR SUPT CREW
        ACT;
        ACT;
      TERM;
;
UTE   AWAIT(3),ALLOC(3);                    CORRECT EXCESSIVE UTE RATE
        ACT,ATRIB(8);                       ACFT MAINTENANCE DELAY
FRAC  EVENT,1;                              FREE ACFT
      TERM;
;
ALTR  AWAIT(4),ALLOC(4);                    REDUCE RESOURCES WHEN AVAIL
      EVENT,12;                             AS DIRECTED BY ENTRY FILE
      TERM;
;
;     THE FOLLOWING CODE IS USED FOR DYNAMIC PRIORITY SWITCHING
;         INCLUDE IT ONLY IF DYNAMIC PRIORITY SWITCHING IS USED
;
;      DETECT,NNQ(1),XP,15,1,1;
;         ACT/5,,XX(2).LE.XX(1),TRM1;
;         ACT/6;
;      EVENT,16;
;TRM1  TERM;
;      DETECT,NNQ(1),XN,10,1,1;
;         ACT/7,,XX(2).GE.XX(1),TRM2;
;         ACT/8;
;      EVENT,16;
;TRM2  TERM;
;
      ENDNETWORK;

INIT,,89.000015;
SIMULATE;                                   START RUN 1
SEEDS,26714690(1),15502262(2),25978034(3),7852978(4),44090237(5),
      91110250(6),32757519(7),97404577(8),20640286(9),33981348(10);
SIMULATE;                                   START RUN 2
SEEDS,68080423(1),1551652(2),78974335(3),35677109(4),96903739(5),
      85250577(6),89409451(7),67260876(8),38187607(9),88543566(10);
SIMULATE;                                   START RUN 3
SEEDS,67133041(1),81338749(2),88356745(3),35973476(4),38163455(5),
      77759315(6),4328327(7),8611515(8),22973375(9),37139174(10);
SIMULATE;                                   START RUN 4
SEEDS,51651670(1),25348811(2),82313721(3),79475719(4),26401394(5),
```

```
     91110513(6),27518256(7),29317783(8),12816531(9),72596993(10);
SIMULATE;                                   START RUN 5
SEEDS,94537255(1),87259859(2),63856401(3),66612547(4),30712585(5),
     69607241(6),37792827(7),148808665(8),66248697(9),51453643(10);
SIMULATE;                                   START RUN 6
SEEDS,35614247(1),182966239(2),185271163(3),46783619(4),57062713(5),
     43886647(6),94107419(7),73849649(8),38244509(9),61157653(10);
SIMULATE;                                   START RUN 7
SEEDS,88319055(1),74863999(2),96908521(3),30258167(4),86689483(5),
     53170445(6),21425047(7),111084483(8),55441121(9),61130261(10);
SIMULATE;                                   START RUN 8
SEEDS,58701573(1),68558063(2),53496517(3),74716657(4),99057583(5),
     33456043(6),42822281(7),45390043(8),54786049(9),15016665(10);
SIMULATE;                                   START RUN 9
SEEDS,92168825(1),36463073(2),47097787(3),80400459(4),94554863(5),
     31567535(6),78212475(7),90560709(8),199227025(9),29923025(10);
SIMULATE;                                   START RUN 10
FIN;
```

Appendix C. Data Files

The model uses six input data files, ten temporary data files, and two output files. Table XXV lists the files used and the logical device numbers assigned to them.

Table XXV. Files Used.

Device	File	Use	Description
1	SOFXX.DAT	Input	XX File.
2	SOFAC.DAT	Input	Aircraft Characteristics
3	SOFWX.DAT	Input	Weather Distribution
4	SOFTG.DAT	Input	Target Locations
5	SOFCD.DAT	Input	SLAM Code
6	CRASOF.OUT	Output	SLAM Summary File
7	-	-	Used by SLAM
8	SOFEN.DAT	Input	Special Entries
9	SOFBS.DAT	Input	Basing File
10-19	-	Temp	Data Files Used by SLAM for RECORD Statements
20	SOFRES.DAT	Output	User Produced Statistics

The files are assigned device numbers external to the model. SLAM RECORD statements were not used in this analysis, but they may be needed for future uses of the model. If they are, the RECORD statements must include ITAPE values between ten and nineteen. The device used for OTPUT special output is twenty, if an output file is needed.

The following pages contain the input data files used in this analysis in alphabetical order.

SOFAC.DAT

```
*****************************************************************************
*                           U N C L A S S I F I E D                         *
*****************************************************************************
*FILENAME=SOFAC  .DAT                                                       *
*NOTE: UPDATE CLASSIFICATION AND FILENAME BEFORE SAVING NEW FILE.           *
*      FILE NAMES: SOFAC__.DAT.  NOTE: USE ONLY 1-99 IN FILE NAME.          *
*      KEEP CHANGES BETWEEN ASTERISK COLUMNS. KEEP DECIMAL POINTS ALLIGNED. *
*****************************************************************************
             AIRCRAFT TYPES           1        2        3        4        5
ITEM#                        *    MC-130   HH-53H  HC-130II HC-130I  CV-22A
*****************************************************************************
 1 ACFT ASSIGNED TO THEATER  *       .00      .00      .00      .00      .00*
 2 CAP/REQ'T (0-3; SEE NOTE) *       .00      .00      .00      .00      .00*
 3 THREAT TYPE (1-3)         *      1.00     1.00     2.00     1.00     1.00*
 4 TOP PRIORITY MSN (0-10)   *      1.00     2.00     5.00     5.00     2.00*
 5 2ND PRIORITY MISSION #    *      3.00     1.00     3.00     3.00     1.00*
 6 3RD PRIORITY MISSION #    *      2.00     3.00     1.00     1.00     3.00*
 7 4TH PRIORITY MISSION #    *       .00     4.00      .00      .00     4.00*
 8 5TH PRIORITY MISSION #    *       .00      .00      .00      .00      .00*
 9 6TH PRIORITY MISSION #    *       .00      .00      .00      .00      .00*
10 7TH PRIORITY MISSION #    *       .00      .00      .00      .00      .00*
11 8TH PRIORITY MISSION #    *       .00      .00      .00      .00      .00*
12 9TH PRIORITY MISSION #    *       .00      .00      .00      .00      .00*
13 10TH PRIORITY MISSION #   *       .00      .00      .00      .00      .00*
14 ATTRITION RATE (%)        *       .10      .10      .10      .10      .10*
15 MECH AIR ABORT (%)        *       .43      .00     2.44     2.44      .00*
16 UTE RATE (HRS/DAY/ACFT)   *      3.00     2.33     2.80     2.80     3.00*
17 SURGE RATE (HRS/DAY/ACFT) *      3.00     2.33     2.80     2.80     3.00*
18 DAYS CAN SUSTAIN SURGE    *       .00      .00      .00      .00      .00*
19 MISSION EFFECTIVENESS (%) *     95.00    95.00    95.00    95.00    95.00*
20 MISSION CAPABLE RATE (%)  *     61.50    58.50    64.50    64.50    72.00*
21 CRASH HAS SURVIVORS       *     15.00    75.00    15.00    15.00    75.00*
22 VTOL CAPABLE (Y=1,N=0)    *       .00     1.00      .00      .00     1.00*
23 AVG CRUISE (KTAS)         *    220.00   120.00   220.00   220.00   220.00*
24 UNREFUELED RADIUS (NM)    *    950.00   290.00  1350.00  1350.00   575.00*
25 REFUEL IN FLIGHT (Y=1,N=0) *      .00     1.00      .00      .00     1.00*
26 REQ'D A/R TRACK (NM)      *       .00    30.00      .00      .00   100.00*
27 RADII BEFORE A/R (0.5-2.0) *      .00     1.50      .00      .00     1.50*
28 BURN RATE (LBS/HR)        *   6000.00  2200.00  6000.00  6000.00  2300.00*
29 MAX FUEL (LBS)            *  59000.00 11800.00 82000.00 82000.00 16500.00*
30 BURN LBS FUEL BEFORE A/R  *       .00      .00      .00      .00      .00*
31 NOT USED                  *       .00      .00      .00      .00      .00*
32 NOT USED                  *       .00      .00      .00      .00      .00*
33 MAX FLY HRS W/O AUGMENTING *     9.00    10.00     9.00     9.00     9.00*
34 CREW RATIO                *      1.50     1.50     1.50     1.50     2.00*
35 AVG ACFT TURN TIME (HRS)  *      1.50     2.00     1.50     1.50     1.00*
```

36 TAKEOFF CEILING MIN (FT)	*	.00	.00	.00	.00	.00*
37 TAKEOFF VIS MIN (SM)	*	.30	.00	.30	.30	.12*
38 ACFT #1 MSN CEILING MIN	*	.00	100.00	1000.00	.00	100.00*
39 ACFT #1 MSN VIS MIN (SM)	*	.00	.25	3.00	1.00	.25*
40 ACFT #1 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
41 RAIN CNX #1 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
42 TURB CNX #1 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
43 ACFT #2 MSN CEILING MIN	*	.00	100.00	1000.00	.00	.00*
44 ACFT #2 MSN VIS MIN (SM)	*	.00	.25	3.00	.00	.00*
45 ACFT #2 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
46 RAIN CNX #2 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
47 TURB CNX #2 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
48 ACFT #3 MSN CEILING MIN	*	300.00	.00	1000.00	.00	.00*
49 ACFT #3 MSN VIS MIN (SM)	*	1.00	.00	3.00	.00	.00*
50 ACFT #3 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
51 RAIN CNX #3 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
52 TURB CNX #3 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
53 ACFT #4 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
54 ACFT #4 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
55 ACFT #4 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
56 RAIN CNX #4 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
57 TURB CNX #4 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
58 ACFT #5 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
59 ACFT #5 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
60 ACFT #5 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
61 RAIN CNX #5 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
62 TURB CNX #5 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
63 ACFT #6 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
64 ACFT #6 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
65 ACFT #6 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
66 RAIN CNX #6 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
67 TURB CNX #6 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
68 ACFT #7 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
69 ACFT #7 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
70 ACFT #7 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
71 RAIN CNX #7 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
72 TURB CNX #7 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
73 ACFT #8 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
74 ACFT #8 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
75 ACFT #8 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
76 RAIN CNX #8 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
77 TURB CNX #8 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
78 ACFT #9 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
79 ACFT #9 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
80 ACFT #9 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
81 RAIN CNX #9 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
82 TURB CNX #9 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*
83 ACFT #10 MSN CEILING MIN	*	500.00	100.00	1000.00	.00	100.00*
84 ACFT #10 MSN VIS MIN (SM)	*	1.00	.25	3.00	.00	.25*
85 ACFT #10 MSN WIND MAX	*	60.00	45.00	60.00	60.00	45.00*
86 RAIN CNX #10 MSN (Y=1,N=0)	*	1.00	1.00	1.00	1.00	1.00*
87 TURB CNX #10 MSN (Y=1,N=0)	*	1.00	.00	1.00	1.00	.00*

```
*******************************************************************************
*                            U N C L A S S I F I E D                          *
*******************************************************************************
NOTE: ITEM 2, CAPABILITY/REQUIREMENTS ENTRY CODE FOLLOWS
              0 - CAPABILITY MODE FOR ACFT & CREW
              1 - REQUIREMENT MODE FOR ACFT; CAPABILITY MODE FOR CREW
              2 - CAPABILITY MODE FOR ACFT; REQUIREMENT MODE FOR CREW
              3 - REQUIREMENT MODE FOR ACFT & CREW
```

If more than five aircraft are used, an additional five can be added by inserting them between the bottom banner and the NOTE.

SOFBS.DAT

```
****************************************************************************
*                         U N C L A S S I F I E D                          *
****************************************************************************
*FILE=SOFBS   .DAT                                                         *
*                                                                          *
*                                                                          *
*                                                                          *
****************************************************************************
* BASE        LOCATION            AIRCRAFT/AIRCREWS ASSIGNED               *
*NUMBER     LAT     LONG     1    2    3    4    5    6    7    8    9   10 *
****************************************************************************
*   1  *  15.188  120.557  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   2  *  26.500  128.500  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   3  *  32.800  129.883  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   4  *  35.900  126.767  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   5  *  25.000  121.500  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   6  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   7  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   8  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   9  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*  10  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   1  *  15.188  120.557  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   2  *  26.500  128.500  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   3  *  32.800  129.883  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   4  *  35.900  126.767  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   5  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   6  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   7  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   8  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*   9  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
*  10  *    .000     .000  .0   .0   .0   .0   .0   .0   .0   .0   .0   .0 *
****************************************************************************
*                         U N C L A S S I F I E D                          *
****************************************************************************
```

The top ten rows are for aircraft assigned, and the bottom ten rows are for aircrews assigned. Assets can be assigned to the theater in this data file or by using entries in SOFEN.DAT to insert them.

SOFEN.DAT

Special entries can be of four types. The first is used to insert aircraft and/or aircrews into theater. The second sets the end of simulation time. The third directs changes in the aircraft capability file during the model run, and the fourth directs changes in the XX array during the model run. The format for the special entries is specified in the file. The FILENAME line must be the first line in the file.

Included in this appendix are the special entry files for all three options and for the quick response option.

OPTION 1

FILENAME=SOFEN1 .DAT

AIRCRAFT TYPE	NUMBER	BASE	CREWS	DATE
1	4	1	6	0
1	3	2	5	0
2	7	4	11	0
3	1	1	2	0
3	2	2	3	0
4	1	1	2	0
4	2	2	3	0
5	3	2	6	0
5	4	3	8	0

END SIMULATION (ALIGN DECIMAL ON COLUMN 49): 89.000015

SPECIAL ENTRIES (ALIGN RIGHT ON COLUMNS 3,8,13; DECIMAL ON 19):

FORMAT FOR ACDATA CHANGE: DATE, AC TYPE, LINE #, VALUE
FORMAT FOR XX( ) CHANGE: DATE, 0, XX NUMBER, VALUE

OPTION 2

FILENAME=SOFEN2 .DAT

AIRCRAFT TYPE	NUMBER	BASE	CREWS	DATE
1	4	1	6	0
1	3	2	5	0
2	3	2	5	0
2	4	4	6	0
3	1	1	1	0
3	1	2	2	0
3	1	3	2	0
4	1	1	2	0
4	1	2	2	0
4	1	3	1	0
5	3	1	6	0
5	2	3	4	0
5	2	4	4	0

END SIMULATION (ALIGN DECIMAL ON COLUMN 49): 89.000015

SPECIAL ENTRIES (ALIGN RIGHT ON COLUMNS 3,8,13; DECIMAL ON 19):

FORMAT FOR ACDATA CHANGE: DATE, AC TYPE, LINE #, VALUE
FORMAT FOR XX( ) CHANGE: DATE, 0, XX NUMBER, VALUE

OPTION 3

FILENAME=SOFEN3 .DAT

AIRCRAFT TYPE	NUMBER	BASE	CREWS	DATE
1	4	1	6	0
1	3	2	5	0
2	7	4	11	0
3	3	1	5	0
4	3	1	5	0
5	3	1	6	0
5	4	2	8	0

END SIMULATION (ALIGN DECIMAL ON COLUMN 49): 89.000015

SPECIAL ENTRIES (ALIGN RIGHT ON COLUMNS 3,8,13; DECIMAL ON 19):

FORMAT FOR ACDATA CHANGE: DATE, AC TYPE, LINE #, VALUE
FORMAT FOR XX( ) CHANGE: DATE, 0, XX NUMBER, VALUE

QUICK RESPONSE OPTION

FILENAME=SOFENQR.DAT

AIRCRAFT TYPE	NUMBER	BASE	CREWS	DATE
1	3	1	5	0
1	2	2	3	0
1	2	5	3	0
2	7	4	11	0
3	1	1	2	0
3	2	2	3	0
4	1	1	2	0
4	2	2	3	0
5	2	2	4	0
5	3	3	6	0
5	2	5	4	0

END SIMULATION (ALIGN DECIMAL ON COLUMN 49): 89.000015

SPECIAL ENTRIES (ALIGN RIGHT ON COLUMNS 3,8,13; DECIMAL ON 19):

FORMAT FOR ACDATA CHANGE: DATE, AC TYPE, LINE #, VALUE
FORMAT FOR XX( ) CHANGE: DATE, 0, XX NUMBER, VALUE

SOFTG.DAT

```
******************************************************************************
*                          U N C L A S S I F I E D                           *
******************************************************************************
*FILE=SOFTG  .DAT                                                            *
*     ALL LATITUDES IN DEGREES N, ALL LONGITUDES IN DEGREES E                *
*     KEEP CHANGES BETWEEN ASTERISKS AND KEEP DECIMALS ALIGNED               *
*     ALL LATITUDES IN DEGREES N, ALL LONGITUDES IN DEGREES E                *
******************************************************************************
* REGION   CUM      NORTHWEST CORNER          SOUTHEAST CORNER      REGION   *
* NUMBER   PROB   LATITUDE  LONGITUDE       LATITUDE  LONGITUDE    PRIORITY  *
******************************************************************************
*    1     .150    23.000   105.000          10.000   110.000        3.00    *
*    2     .250    20.000    95.000          10.000   105.000        6.00    *
*    3     .400    30.000   110.000          22.000   120.000        5.00    *
*    4     .500    33.000   117.000          30.000   122.000        2.00    *
*    5     .800    42.000   115.000          35.000   125.000        1.00    *
*    6    1.000    44.000   125.000          38.000   131.000        4.00    *
*    7    1.000     0.000     0.000           0.000     0.000        7.00    *
*    8    1.000     0.000     0.000           0.000     0.000        8.00    *
*    9    1.000     0.000     0.000           0.000     0.000        9.00    *
*   10    1.000     0.000     0.000           0.000     0.000       10.00    *
*    1     .150    23.000   105.000          10.000   110.000        1.00    *
*    2     .250    20.000    95.000          10.000   105.000        2.00    *
*    3     .400    30.000   110.000          22.000   120.000        3.00    *
*    4     .500    33.000   117.000          30.000   122.000        4.00    *
*    5     .800    42.000   115.000          35.000   125.000        5.00    *
*    6    1.000    44.000   125.000          38.000   131.000        6.00    *
*    7    1.000     0.000     0.000           0.000     0.000        7.00    *
*    8    1.000     0.000     0.000           0.000     0.000        8.00    *
*    9    1.000     0.000     0.000           0.000     0.000        9.00    *
*   10    1.000     0.000     0.000           0.000     0.000       10.00    *
******************************************************************************
*                          U N C L A S S I F I E D                           *
******************************************************************************
```

The top ten regions are for SOF, and the bottom ten regions are for Combat Rescue.

SOFWX.DAT

Two weather files are included here, one for winter and one for summer.

WINTER

```
************************************************************************
*                       U N C L A S S I F I E D                       *
************************************************************************
*FILENAME=SOFWX1 .DAT  WINTER                                          *
*NOTE: UPDATE CLASSIFICATION AND FILENAME BEFORE SAVING NEW FILE.      *
*      FILE NAMES: SOFWX__.DAT.  USE ONLY 1-99 IN FILE NAMES.          *
*      KEEP CHANGES BETWEEN ASTERISK COLUMNS. KEEP DECIMAL POINTS ALLIGNED. *
************************************************************************
```

REGIONAL WEATHER PROBABILITIES FOR ARRAY WXDATA()

	CEILING(FT) COL 1-6:						TURBULENCE(MOD+) COL 7
COL#	1	2	3	4	5	6	7
VAL<	100.00	300.00	500.00	1000.00	1500.00	5000.00	1.00
HOME 1*	.05	.50	1.00	2.00	3.00	27.00	.05*
HOME 2*	.06	.50	2.00	6.00	14.00	29.00	.05*
HOME 3*	.05	.60	3.00	9.00	28.00	38.00	.05*
HOME 4*	.00	.60	1.00	2.00	4.00	36.00	.40*
HOME 5*	.05	.50	1.00	2.00	3.00	27.00	.05*
HOME 6*	.00	.00	.00	.00	.00	.00	.00*
HOME 7*	.00	.00	.00	.00	.00	.00	.00*
HOME 8*	.00	.00	.00	.00	.00	.00	.00*
HOME 9*	.00	.00	.00	.00	.00	.00	.00*
HOME 10*	.00	.00	.00	.00	.00	.00	.00*
AREA 1*	.16	.38	.96	2.07	5.36	27.00	.20*
AREA 2*	.16	.45	.85	.95	2.21	12.50	.15*
AREA 3*	.16	.42	1.05	3.56	7.25	30.05	.20*
AREA 4*	.21	.35	.86	1.89	2.75	11.64	.18*
AREA 5*	.30	.52	.76	2.85	4.09	14.63	.17*
AREA 6*	.35	.59	.81	3.18	4.80	16.74	.16*
AREA 7*	.00	.00	.00	.00	.00	.00	.00*
AREA 8*	.00	.00	.00	.00	.00	.00	.00*
AREA 9*	.00	.00	.00	.00	.00	.00	.00*
AREA 10*	.00	.00	.00	.00	.00	.00	.00*

VISIBILITY(SM) COL 8-11:					WIND(KN) COL 12-15:				RAIN(HEAVY) COL 16
COL#	8	9	10	11	12	13	14	15	16
VAL<	.25	1.00	2.00	3.00	13.00	20.00	35.00	45.00	1.00
HOME 1*	.05	.30	1.00	2.00	98.00	99.50	99.95	99.99	.20*
HOME 2*	.05	.50	4.00	5.00	97.50	99.65	99.95	99.99	.15*
HOME 3*	.05	.80	6.00	10.00	97.00	99.70	99.95	99.99	.30*
HOME 4*	.70	2.00	5.00	8.00	80.00	89.00	95.00	99.99	.10*
HOME 5*	.05	.30	1.00	2.00	98.00	99.50	99.95	99.99	.20*
HOME 6*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 7*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 8*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 9*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 10*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 1*	.21	.76	2.30	12.60	94.73	97.26	99.50	99.70	.20*
AREA 2*	.62	1.71	4.80	15.86	89.20	96.15	99.56	99.90	.10*
AREA 3*	.25	.98	4.61	17.72	93.78	98.55	99.20	99.95	.15*
AREA 4*	.29	1.16	5.20	17.99	91.50	97.56	98.78	99.90	.16*

AREA 5*	.27	.89	2.05	7.65	92.18	97.79	99.90	99.99	.10*
AREA 6*	.32	.98	2.14	8.46	90.89	96.34	99.83	99.95	.12*
AREA 7*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 8*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 9*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 10*	.00	.00	.00	.00	.00	.00	.00	.00	.00*

****

*   U N C L A S S I F I E D   *

****

SUMMER

```
******************************************************************************
*                          U N C L A S S I F I E D                           *
******************************************************************************
*FILENAME=SOFWX2 .DAT   SUMMER                                               *
*NOTE: UPDATE CLASSIFICATION AND FILENAME BEFORE SAVING NEW FILE.            *
*      FILE NAMES: SOFWX__.DAT.  USE ONLY 1-99 IN FILE NAMES.                *
*      KEEP CHANGES BETWEEN ASTERISK COLUMNS. KEEP DECIMAL POINTS ALLIGNED.  *
******************************************************************************
```

REGIONAL WEATHER PROBABILITIES FOR ARRAY WXDATA()

	CEILING(FT) COL 1-6:						TURBULENCE(MOD+) COL 7
COL#	1	2	3	4	5	6	7
VAL<	100.00	300.00	500.00	1000.00	1500.00	5000.00	1.00
HOME 1*	.05	.50	1.00	2.00	3.00	27.00	.05*
HOME 2*	.07	.55	2.00	6.00	20.00	31.00	.05*
HOME 3*	.10	.60	3.00	9.00	28.00	38.00	.05*
HOME 4*	.10	2.00	4.00	8.00	12.00	32.00	.05*
HOME 5*	.05	.50	1.00	2.00	3.00	27.00	.05*
HOME 6*	.00	.00	.00	.00	.00	.00	.00*
HOME 7*	.00	.00	.00	.00	.00	.00	.00*
HOME 8*	.00	.00	.00	.00	.00	.00	.00*
HOME 9*	.00	.00	.00	.00	.00	.00	.00*
HOME 10*	.00	.00	.00	.00	.00	.00	.00*
AREA 1*	.00	.09	.32	.59	8.00	32.40	.15*
AREA 2*	.00	.14	.37	1.22	8.00	30.00	.05*
AREA 3*	.02	.17	.52	1.34	8.65	33.00	.10*
AREA 4*	.01	.15	.45	1.25	8.25	32.50	.08*
AREA 5*	1.05	2.86	8.55	9.15	10.25	35.85	.04*
AREA 6*	1.13	3.43	9.38	10.78	12.22	40.07	.05*
AREA 7*	.00	.00	.00	.00	.00	.00	.00*
AREA 8*	.00	.00	.00	.00	.00	.00	.00*
AREA 9*	.00	.00	.00	.00	.00	.00	.00*
AREA 10*	.00	.00	.00	.00	.00	.00	.00*

	VISIBILITY(SM) COL 8-11:				WIND(KN) COL 12-15:				RAIN(HEAVY) COL 16
COL#	8	9	10	11	12	13	14	15	16
VAL<	.25	1.00	2.00	3.00	13.00	20.00	35.00	45.00	1.00
HOME 1*	.05	.30	1.00	2.00	98.00	99.50	99.95	99.99	.20*
HOME 2*	.08	.50	3.00	7.00	97.50	99.55	99.95	99.99	.20*
HOME 3*	.10	.80	6.00	10.00	97.00	99.70	99.90	99.99	.30*
HOME 4*	.30	2.00	4.00	7.00	92.00	97.00	99.90	99.99	.50*
HOME 5*	.05	.30	1.00	2.00	98.00	99.50	99.95	99.99	.20*
HOME 6*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 7*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 8*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 9*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
HOME 10*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 1*	.28	.31	.73	2.97	97.38	99.25	99.90	99.95	1.13*
AREA 2*	.20	.30	1.14	2.95	98.15	99.28	99.85	99.90	.76*
AREA 3*	.23	.35	1.20	3.14	98.25	99.35	99.85	99.99	1.05*

AREA 4*	.20	.28	.98	2.88	98.50	98.85	99.87	99.95	.85*
AREA 5*	.95	1.05	6.57	15.85	97.55	99.65	99.90	99.99	.25*
AREA 6*	1.01	1.15	7.45	19.52	96.80	99.40	99.50	99.90	.32*
AREA 7*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 8*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 9*	.00	.00	.00	.00	.00	.00	.00	.00	.00*
AREA 10*	.00	.00	.00	.00	.00	.00	.00	.00	.00*

****

* U N C L A S S I F I E D *

****

SOFXX.DAT

```
*************************************************************************
*                          U N C L A S S I F I E D                      *
*************************************************************************
*FILENAME=SOFXX  .DAT                                                   *
*NOTE: UPDATE CLASSIFICATION AND FILENAME BEFORE SAVING NEW FILE.       *
*      FILE NAMES: SOFXX__.DAT. NOTE: USE ONLY 1-99 IN FILE NAMES.      *
*      CHANGES LIMITED TO 7 CHARACTERS INCLUDING DECIMAL POINTS AS ALLIGNED. *
*************************************************************************
```

XX()	1	2	3	4	5	6	7	8	9	10
1- 10	1.00	2.00	3.00	4.00	.00	.00	.00	.00	.00	.00
11- 20	.00	99.00	10.00	5.00	99.00	5.00	10.00	5.00	.00	.00
21- 30	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
31- 40	.88	.87	.88	.87	.88	.87	.00	.00	.00	.00
41- 50	9.00	9.00	9.00	9.00	9.00	9.00	.00	.00	.00	.00
51- 60	5.00	5.00	5.00	5.00	5.00	5.00	.00	.00	.00	.00
61- 70	7.00	7.00	7.00	7.00	7.00	7.00	.00	.00	.00	.00
71- 80	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00
81- 90	99.00	25.00	10.00	.00	16.00	3.00	108.00	12.00	.00	.00
91-100	475.00	415.00	500.00	.00	.00	.00	.00	.00	.00	.00
101-110	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
111-120	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
121-130	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
131-140	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
141-150	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
151-160	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
161-170	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
171-180	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
181-190	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
191-200	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
201-210	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
211-220	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
221-230	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
231-240	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
241-250	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
251-260	20.00	10.00	1.00	.00	.00	-4.00	-20.00	67.00	63.00	59.00
261-270	55.00	51.00	47.00	43.00	39.00	35.00	31.00	.00	.00	.00
271-280	.00	.00	.00	.00	.00	.00	2.00	5.00	5.00	.00
281-290	99.00	99.00	3.00	4.00	23.00	2.33	4.00	1.57	.80	.00
291-300	1.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
301-310	.50	.30	.30	.60	.75	.80	.00	.00	.00	.00
311-320	1.00	1.00	1.00	1.00	1.00	1.00	.00	.00	.00	.00
321-330	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
331-340	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
341-350	.00	.00	.00	.00	1.00	.00	.00	.00	.00	.00
351-360	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
361-370	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

371-380	.00	.00	.00	.00	.00	.00	.00	.00	.00	12.00
381-390	60.00	60.00	60.00	2.00	6.00	.00	.00	.00	.00	.00
391-400	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

****
*   U N C L A S S I F I E D   *
****

BIBLIOGRAPHY

1. Capicik, LtCol Paul J. Telephone Interview. Headquarters Military Airlift Command (Hq MAC/XONP), Scott AFB, IL, July 8, 1986.

2. Daskin, Mark S. "A Maximum Expected Covering Location Model: Formulation, Properties and Heuristic Solution," Transportation Science, 17: 48-70 (February 1983).

3. Dickinson, Maj Jack R. "Combat Rescue and Special Operations Forces Model Programmer's Manual," Headquarters Military Airlift Command (Hq MAC/XPSS) September 1, 1985.

4. Niemeyer, Capt Joseph B. Personal Correspondence. Studies Division, Headquarters Military Airlift Command (Hq MAC/XPCS), Scott AFB, IL, July 30, 1986.

5. SAS Institute, Inc. SAS Introductory Guide, Third Edition. Cary, NC: SAS Institute, 1985.

6. Sherali, Hanif D. and Warren P. Adams. "A Decomposition Algorithm for a Discrete Location-Allocation Problem," Operations Research, 32: 878-900 (July/August 1984).

7. Thisse, J. -F., J. E. Ward, and R. E. Wendell. "Some Properties of Location Problems with Block and Round Norms," Operations Research, 32: 1309-1327 (November/December 1984).

8. Weaver, Jerry R. and Richard L. Church. "Computational Procedures for Location Problems on Stochastic Networks," Transportation Science, 17: 168-180 (May 1983).

9. -----. "A Median Location Model with Nonclosest Facility Service," Transportation Science, 19: 58-74 (February 1985).

VITA

Captain Mark E. Kraus was born on 12 January in New Brunswick, New Jersey. He graduated from high school in Hendersonville, North Carolina in 1977. In June 1982, he graduated from the United States Air Force Academy with a Bachelor of Science degree with a major in Mathematics. Upon graduating, he was commissioned a Second Lieutenant in the USAF. He was assigned to Headquarters, Aerospace Rescue and Recovery Service as an Operations Analyst. When the Twenty-third Air Force was activated in February 1983, he was assigned as the 23rd Air Force staff analyst. In that position, he became involved with AFSOF force sizing. In June 1985, Capt Kraus entered the School of Engineering at the Air Force Institute of Technology. He is a member of Omego Rho.

Permanent address: P.O. Box 27

Sterling, NY 13156

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

A179 422

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release;
distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
AFIT/GOR/OS/86D-6

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
School of Engineering

6b. OFFICE SYMBOL *(If applicable)*
AFIT/ENS

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State and ZIP Code)*
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433

7b. ADDRESS *(City, State and ZIP Code)*

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State and ZIP Code)*

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.

11. TITLE *(Include Security Classification)*
See Box 19

12. PERSONAL AUTHOR(S)
Mark E. Kraus, Captain, USAF

13a. TYPE OF REPORT
MS Thesis

13b. TIME COVERED
FROM ________ TO ________

14. DATE OF REPORT *(Yr., Mo., Day)*
1986 December

15. PAGE COUNT
160

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.
15	07	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Rescue, Simulation, Deployment, Strategy, Special

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Title: BASING THE US AIR FORCE SPECIAL OPERATIONS FORCES

Thesis Advisor: John R. Valusek, Lt Colonel, USAF

Approved for public release: IAW AFR 190-17.
Lynn E. Wolaver 7 April 87
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
John R. Valusek, Lt Colonel, USAF

22b. TELEPHONE NUMBER *(Include Area Code)*
(513) 255-3362

22c. OFFICE SYMBOL
AFIT/ENS

The Air Force Special Operations Forces (AFSOF) are currently overtasked. The only existing Hq MAC AFSOF model is limited by the assumption that all AFSOF assets are colocated. This research ~~effort~~ was directed toward removing this limitation so the model could be used to address basing questions. The model was modified to include geographical locations rather than just distances. The target data base was modified, and a basing data base was added.

The model was then demonstrated using a representative scenario and representative data. The scenario and data were coordinated with Hq MAC to insure they were of the right form but not close to any classified information. The study involved three basing options to be compared for mission accomplishment. The options were compared for total successful missions, ~~which was~~ also broken down by mission type; average mission delay by mission type; and how well they implemented the desired regional priority scheme.

The uses and limitations of the model as well as potential areas of improvement were discussed. A major limitation of the model is its restriction to use for long range planning. A look was taken ~~at~~ a deterministic model that could provide a short time response. It appeared feasible to use location/allocation methods. Such a model could be used in a decision support system to provide real time help in basing the AFSOF.

END
5_87
DTIC